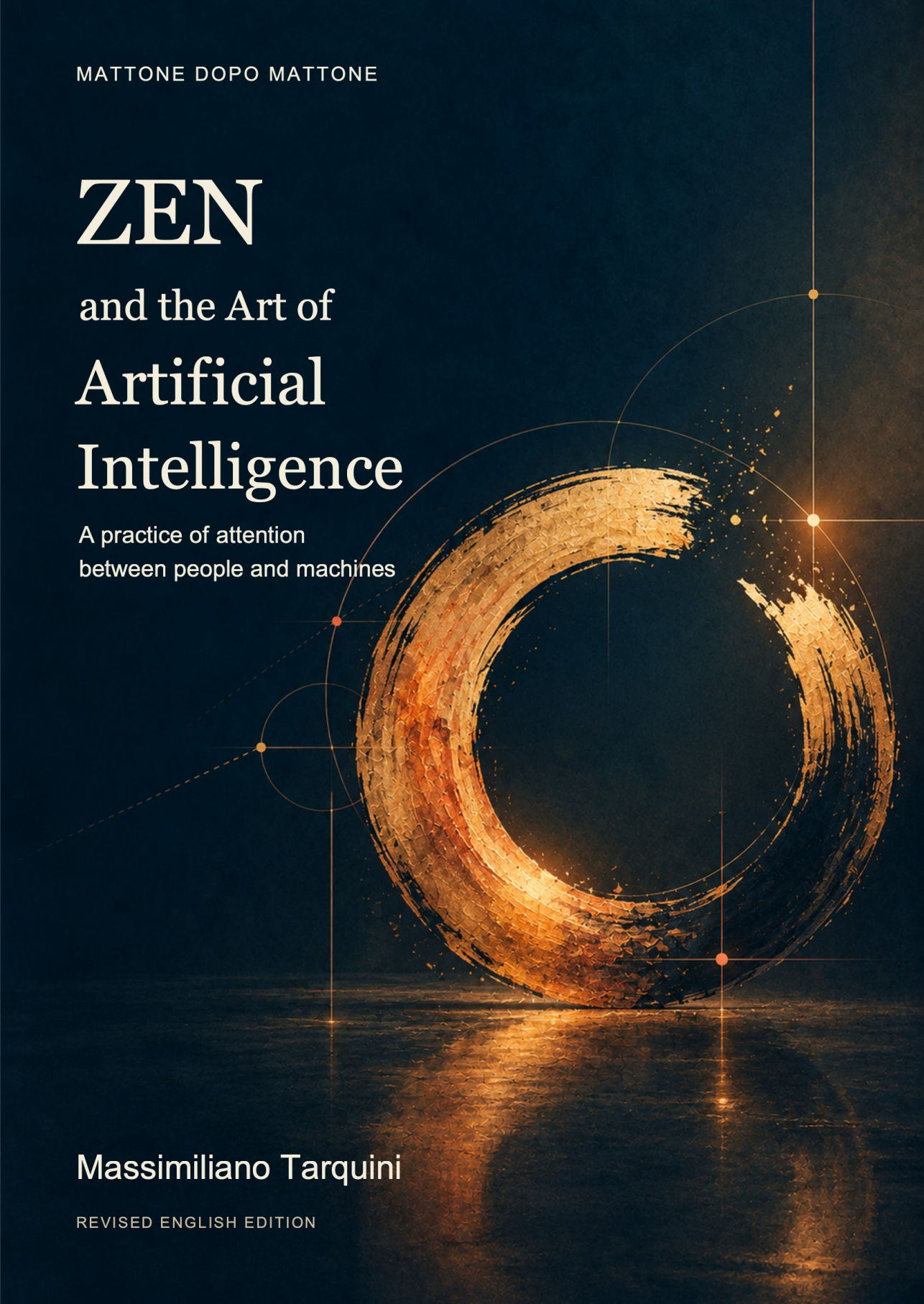
MATTONE DOPO MATTONE

ZEN
and the Art of Artificial Intelligence

A practice of attention
between people and machines



Massimiliano Tarquini

REVISED ENGLISH EDITION

ZEN and the Art of Artificial Intelligence

A practice of attention between people and machines

Massimiliano Tarquini

Mattone dopo mattone

Revised English edition · October 2026

Contents

Introduction Working with attention 5

Prologue At the edge of silicon 7

Part I Before believing the machine 10

1 What we call artificial intelligence 12

2 How a model learns 21

3 When language becomes an interface 31

Part II Between capabilities and illusions 44

4 The capabilities that emerge 46

5 Plausible answers facts and hallucinations 53

6 The world in the data bias and representation 58

Part III AI in the fabric of life 66

7 A mirror of culture 68

8 Small everyday acts of delegation 74

9 Small is beautiful intelligence close to people 80

Part IV The practice of attention 85

10 Pausing before asking 87

11 Reading an answer without clinging to it 91

12 Collaborating while retaining your judgement 95

Part V Designing with presence 99

13 Human-centred artificial intelligence 101

14 From intention to design choices 105

15 Teaching and learning to live with AI 110

Epilogue Code as a kōan 115

Appendix A A glossary to explore 119

Appendix B Questions and annotated practices 124

Appendix C Reading and references 128

Appendix D List of figures 131

The author 132

Introduction Working with attention

I have found something of Zen in working with language models. This book grows out of that experience.

The word needs some clarification. I am not suggesting that the machine possesses a hidden wisdom, or turning a conversation on a screen into a spiritual practice. I am speaking about the way this work invites me to pay attention: to the question, to the answer, and to what I do when the two fail to meet.

A model returns words with an ease that can encourage us to hurry. A request, an answer, another request. The result always seems close at hand. Yet within that very movement, it becomes possible to notice something: I had not made clear what I was looking for; I had left a condition unstated; I was asking for reassurance instead of trying to understand.

The first change, then, concerns the question. Before sending it, I can pause long enough to recognise the problem. What do I want to achieve? Which part of the context have I actually shared? How will I know whether the result is useful? The pause does not make the request perfect. It gives it a direction and makes its shortcomings visible.

The second change concerns reading. A well-written answer can bring relief: at last, someone has given shape to a confused thought. That relief deserves attention, because it is easy to turn it into trust. I can appreciate a sentence and still ask whether it is correct. I can recognise a good proposal and decide not to use it. I can find an error without rejecting everything the interaction has helped me see.

This is where I recognise an affinity with Zen: in the possibility of staying with what happens without immediately clinging to a conclusion. The model does not have to be right for me to learn something from its answer. Some-

times an answer that misses the point makes what I wanted to say clearer. This does not excuse the error; it changes the way I can work through it.

Of course, attention does not replace checking. A reference must be opened and read. A calculation must be done again. A choice that involves other people requires us to listen to them. Verification does not interrupt this way of working: it belongs to it. It is the act through which I bring words back to what they ought to describe.

I am also interested in what remains after the answer. If the text is convincing, can I still explain it in my own words? If I like a proposal, can I say why? If the machine has written a great deal, have I kept enough room for my own judgement? The work becomes valuable when my ability to see grows alongside the result, rather than when the result merely allows me to skip over the problem.

This book follows that movement. It enters the mechanisms of models to remove mystery where precision is needed. It then returns to the experience of interaction, where precision alone cannot answer every question. It discusses culture, delegation, responsibility and design because generated words do not always remain on the screen: they enter our documents, our decisions and our relationships.

The title, *ZEN and the Art of Artificial Intelligence*, gives shape to the open path from which the book emerged. I hope these pages help readers approach technology with curiosity and work with greater clarity, without losing the rhythm of thought. Ask a question. Listen to an answer. Check it. Look again.

Massimiliano Tarquini

Prologue At the edge of silicon

There is a moment, just before fingers touch the keyboard, when the question exists only for the person about to write it. The screen is on, the machine is available, and breath accompanies an intention that is not yet fully formed. At this threshold, something ancient, the human need to give shape to a thought, meets something recent: a system that can return words through calculations and learned representations.

We can call this relationship the edge of silicon. It does more than separate a person from a device: it brings speed into contact with stillness, technical capability with human meaning, what the system can produce with what we choose to recognise in it. AI writes with us, transforms material and answers in familiar tones; this is precisely why it can surprise us while also encouraging us to attribute something to the answer that we have not yet verified.

Crossing that boundary requires two kinds of clarity. The first concerns mechanisms: distinguishing training from use, a model from a service, context from memory. The second concerns our relationship with those mechanisms: recognising the expectations we bring to a question, the relief an answer may offer, and our responsibility for what we will do with the text. A mathematical description does not exhaust this experience, while a metaphor cannot replace the mathematical description.

Why Zen

The book approaches Zen by acknowledging its origins in Buddhist traditions and the depth of a practice it does not claim to represent. It draws from this encounter a personal way of attending: the possibility of observing what happens before enclosing it in a definition, recognising an assumption, and leaving room for what we do not know. This choice belongs to the voice of the book; it does not turn a statistical model into a spiritual reality.

The ensō, a circle drawn with a brush and left open, gives form to this orientation. In a setting that encourages prediction, optimisation and completion, the unfinished stroke reminds us that understanding does not always mean filling every space. We can approach a system precisely and keep asking how it changes what we see, write and choose.

This is where the journey begins: neural networks and contemplative images, requests and evidence, culture and responsibility, leading eventually to design decisions. You need not be a computer scientist to follow it, but you do need a willingness to examine one step before using it as the foundation for the next. The method of the *Mattone dopo mattone* series, brick by brick, thus enters this unconventional book as well: building an understanding that enables the reader to choose, as well as to recognise words.

A word of respect

The images of emptiness, echo, mirror and no-mind are analogies proposed in these pages. They neither describe the model technically nor amount to Zen teaching or a path of practice. Where a statement has no identified traditional source, it is presented as a reflection of the book; for a description of zazen practice, we refer readers to a source within the Sōtō tradition. [R9]

The reference to Zen remains a respectful encounter, rather than a claim to represent all its richness. Any inaccuracies belong to the book and must be open to correction. The same willingness shapes our relationship with AI: curiosity becomes valuable when it remains open to verification and to listening to someone who can show us a limit we had not seen.

Finding your way through these pages

The first part builds the vocabulary needed to approach the system. The second distinguishes capability, plausibility and fairness; the third brings those distinctions into everyday life. The fourth considers how to ask, read and collaborate, while the fifth turns attention into choices about design and

teaching. The appendices offer terms, annotated practices and further reading, allowing us to return to steps that call for a pause.

The prose carries the argument, while the boxes explore a mechanism or a limitation in greater depth. Sample responses are illustrative unless a reproducible test is specified; in the activities, a worked case and a variation allow us to see whether a criterion can be adapted. The book is intended for people who study, teach, design, work or take part in civic life, and who want to understand how to use these possibilities without losing their own judgement.

You do not need prior knowledge of neural networks. When a calculation appears, we first read it as a sequence of actions, then substitute numbers and check the result. When we meet an illustrative answer, we observe what it preserves, adds or leaves out. You can pause at one of these steps and try explaining it in your own words: if a connection is missing, return to the case that makes it concrete.

We can begin with a question: what happens when silicon echoes our words and we begin to listen? The answer also concerns how we will be present as the text arrives, and what we choose to do once it has arrived.

Part I Before believing the machine



The circle, the lotus and the wave accompany us at the beginning of the journey. The ensō circle remains open and invites us not to close too soon what we are observing; the lotus rises from a bed we cannot see and reminds us that a clear form may emerge from complex conditions; the wave suggests a movement to follow attentively, without turning every step into a conquest.

We can bring these images into our first question about AI. The answer we read is the visible part of a system: behind its fluency are data, parameters, procedures and choices. Before believing or rejecting it, we need to approach these steps with an adequate vocabulary. We do so not to remove every surprise, but to recognise which surprise belongs to the mechanism and which to the way we encounter it.

We shall first distinguish who does what within the service: the model generates words, while other components may retrieve a document or perform a calculation. Following photographs of apples and oranges, we shall then see how weights change during learning and how to test the result on new pictures. Finally, we shall enter the request and the transformations that make a continuation possible. The images keep open a question that definitions alone cannot exhaust: what do we choose to attribute to the machine when it echoes our words?

Chapter 1 What we call artificial intelligence

You open a browser, write a question and wait. A little later, the screen fills with text that has an order, a voice and a direction: the answer picks up your words, distinguishes several points and seems to understand where you wanted to go. Even before judging its content, you may easily feel the presence of someone to talk to. This impression should not be dismissed, because it tells us something about our relationship with language; it does, however, need to be understood if we want to use well the tool that produced it.

Let us begin at this threshold, rather than with an abstract definition. What do we recognise as intelligence when a machine answers us? Perhaps the ability to find a relationship we had not seen, explain a concept or adapt to the tone of a question. All these performances can be useful and, in some cases, surprising. The step that requires attention comes when we infer from a performance something we have not observed: an inner experience, an intention, a care directed towards us.

The image of the cup to be emptied can accompany us here. It asks us not to give up knowledge, but to make room for distinctions that habit may erase. The words "learn", "see", "remember" and "decide" have a long human history; in computing they also describe technical functions. Understanding AI means learning to move between these uses, without assuming that the same verb denotes the same process.

The machine that seems to think

The human brain developed through evolution and is inseparable from a body, emotions, memories and relationships. An artificial neural network, by contrast, is a mathematical structure: it combines numerical values through weights, sums and transformation functions. The name "neural" recalls an inspiration, rather than a biological identity. If the system writes a sentence

about pain, the observable fact is that it has produced a sentence; this does not demonstrate that it feels pain.

This distinction does not diminish what the system can do. A solution may be correct even if the process producing it differs from ours. A text may help us understand a problem even if the machine does not share our experience. The question of artificial understanding remains a subject of debate: meanwhile, to work clearly, we can distinguish observable capabilities from human experience, avoiding the inference of the latter from the presence of the former alone.

When we ask an assistant what the weather will be like, we hear a voice answering and tend to imagine a subject behind it. Yet the service may have recognised the request, consulted a weather source and composed a sentence: these are coordinated functions that we should evaluate separately. Did it identify the location? Did it use current data? Did it report the forecast without altering it? These questions are more useful than a general impression of intelligence, because they tell us where to check.

The model within the service

Three different words. AI is a broad collection of methods for performing tasks that require capabilities such as recognition, prediction and planning. A model is a structure adjusted through data that transforms inputs into outputs. A service combines models, an interface, documents, tools and rules. Saying "AI searched" may describe the service without meaning that the model's parameters consulted a web page on their own.

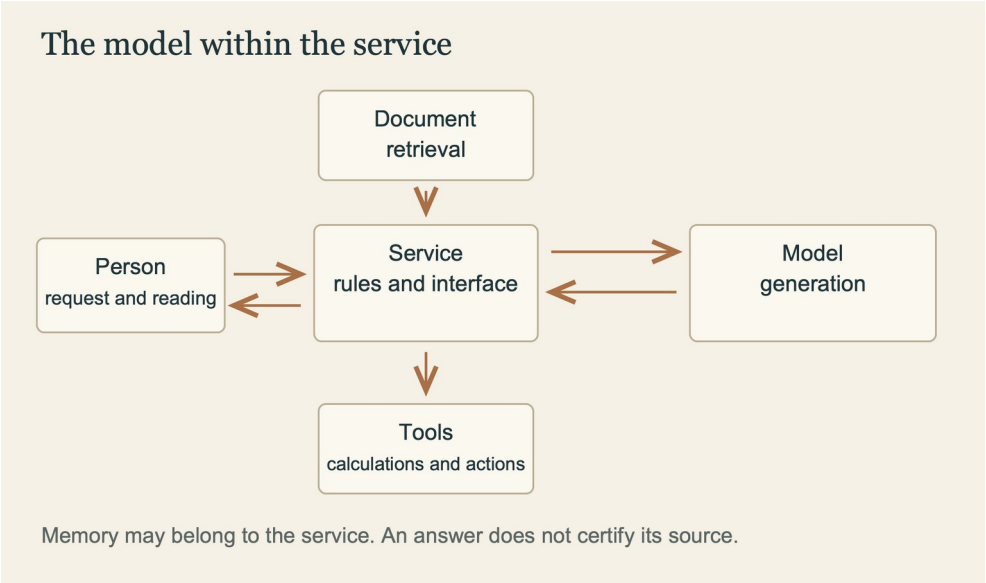


Figure 1 · The model is one component of the service

In the figure, the user encounters a service: the model generates, a repository may supply documents, and a tool may carry out a calculation. The arrows show exchanges of information. If the service retains a preference, we ask where it is stored and when it is used; if it cites a page, we ask whether that page was actually retrieved or merely named during generation.

A language model, considered at the moment of generation, receives an input and calculates a probability distribution for the next token. A token is a unit used in text processing that may correspond to a word or part of a word; we will meet it more closely in Chapter 3. A procedure selects an alternative, and the process continues, also using what has just been generated. Selection does not always mean choosing the most probable token: it may involve sampling, choosing among alternatives according to a distribution.

We can compare the model to a bell that remains silent until struck. The image makes clear that an answer requires an execution, but it has a limit: a service can start scheduled executions, retrieve conversations, search for materials and coordinate several calls. The model is one component of that service, rather than everything behind the screen. To know what is stored or

started, we must understand the actual system instead of attributing every behaviour of the application to a single mathematical function.

A check to take with you

Suppose an assistant answers a question about the weather: the result is current only if the service has up-to-date evidence and uses it correctly. Look for the location, observation time and source, because a natural tone does not make old data current. The same distinction transfers to a question about a shop's opening hours: a model may formulate the answer, but the information must come from a relevant, current source.

This criterion is useful whenever the question depends on the present world, rather than solely on a transformation of supplied text. If the source is missing, we have recognised the answer's limit and can look elsewhere. Clarity begins here: knowing which part of the work has been done and which remains to be done.

Knowing something and checking it

If we ask who painted the Mona Lisa and receive "Leonardo da Vinci", the correct answer can make the model seem like an encyclopaedia available for conversation. Factual information may indeed be represented in its parameters, and some fragments of training data may be memorised. Research has documented cases of data extraction from language models: it would therefore be inaccurate to say that a model never retains specific information. [R2] A decisive distinction remains, however: producing an answer does not, by itself, verify its truth.

> **What research on memorisation shows.** Carlini and colleagues studied the possibility of extracting some training texts from a model. [R2] Their work helps explain why a familiar answer may also depend on specific information retained in parameters. It does not establish that every answer is copied or that all models retain the same data.

We can understand this by observing the completion "The quick brown fox…", which may continue with "...jumps over the lazy dog". The familiarity of the sequence helps produce its continuation, without requiring the system to see a fox or recall a scene it has experienced. In real tasks, the patterns are much more complex, and the representations can support translations, explanations and logical steps. What remains unchanged is the need to check the relationship between the sentence produced and what it ought to describe.

A model can prepare a wedding speech, a separation letter or a philosophical reflection: the text may even move us, because language carries traces of human experiences. The reader's emotion is real, but it does not demonstrate emotional participation by the machine. Likewise, the precision of one explanation does not show that every subsequent explanation will be precise. Fluency makes the text readable; reliability requires another kind of work.



Figure 2 · The sound of one hand: in this illustrative scene, the student asks the question and the master remains attentive. The pause invites us to observe the question itself.

In the illustration, the student asks the master what the sound of one hand is. Silence may change the question. A model, on the other hand, may produce an explanation or a poetic answer because we have asked it to generate. We do not have to decide which is wiser to recognise that they are doing different kinds of work. The distinction helps us read the result for what it offers: a linguistic possibility to evaluate, rather than an authority to follow merely because it can speak.

The world enters the data

A machine does not become neutral because it uses numbers. Data have been collected, selected, classified and made available through choices; training objectives and service rules also express priorities. If language represents certain people more often in particular roles, a system may reproduce those associations. The orderly appearance of the answer may then conceal a distortion already familiar to us in the social world.

Imagine the request "Describe someone who is a successful entrepreneur". If the answer automatically introduces a man, we can ask what information in the prompt justifies that gender and what has instead been added. An isolated example does not measure all the system's fairness, but it makes visible a choice that might otherwise go unnoticed. In Chapter 6 we will widen our view from individual terms to data, metrics and consequences.



Figure 3 · A partial mirror: the result retains traces of culture, but selects and transforms them.

The metaphor of the mirror helps us recognise a relationship: in the result we also encounter the culture from which the system was built. However, a literal mirror reflects what stands before it, while a model selects and generates; it does not produce a faithful copy of the world. The image invites us to look for shadows, provided it does not make us forget the mechanism that transforms them.

The work we choose to delegate

The question "Will AI replace us?" is too broad to receive a single answer. Some tasks can be automated, others transformed, while still others remain bound to responsibilities and contexts that a generated answer cannot assume on our behalf. Work is more than a collection of operations: it includes criteria, relationships, commitments and the ability to answer for consequences. It is therefore more useful to ask which part we are delegating and how we will know whether the delegation works.

A model may help us prepare a CV, but it does not determine on its own whether the job matches our circumstances and aspirations. A system may

follow a procedure for cooking an omelette, whereas noticing what a person needs on a difficult day requires a different relational context. These examples do not establish a permanent boundary for technical capabilities: they show that a bounded performance does not exhaust the human meaning of an activity.

Collaboration becomes interesting when it allows us to do better work whose quality we can still judge. A document summary may save time, but the person using it must be able to see which conditions have been retained or omitted. Generated code may offer a starting point, but its behaviour must be understood and tested. Having the result is not enough: we need to know which check makes it usable.

The stories through which we encounter machines

Long before contemporary services, literature and cinema had given shape to our expectations. In *Frankenstein*, published by Mary Shelley in 1818, we encounter a creature born of ambition and then abandoned: the story directs our attention to the creator's responsibility and the created being's need for relationship. It is not a technical model of AI, but a question about care that can accompany any powerful project.

Through HAL 9000, in the 1968 film *2001 A Space Odyssey*, we can read the risk of incompatible constraints and of a logic given insufficient room for doubt. This is an interpretation of the story, rather than a diagnosis of real systems. In *Her*, from 2013, Samantha's voice instead explores the desire to be listened to and reflected: what seems like intimacy invites us to ask how much depends on the interlocutor and how much on our expectations.

These stories resonate because the machine becomes a lens on human identity. When a creation reflects us, we may feel wonder, fear or closeness; none of these reactions, on its own, describes how that creation works. The book makes room for stories precisely to hold the two levels together: understanding the system and understanding how we encounter it.

The clarity of a distinction

The risk of attributing almost magical powers to technology coexists with the opposite risk of believing that, once matrix multiplication has been named, nothing remains to be understood. The mathematical description explains a mechanism; the experience of use reveals effects on work, language and people. An attentive view needs both, because a formula does not cancel a consequence, and a consequence does not demonstrate an inner experience in the model.

The first step is therefore to loosen assumptions, as we empty a cup before filling it. We can recognise a capability without imagining a person, appreciate an answer without turning it into evidence, and delegate a task without losing judgement over its result. The weather check has shown us how to distinguish a plausible sentence from verified information. In the next chapter we shall follow what changes in a model during training, including a small calculation with its weights.

Chapter 2 How a model learns

The word "learning" brings familiar images with it: a person tries something, notices an error, remembers an experience and changes how they act. When we use it for a machine, some of these images help and others can lead us astray. The model changes, but we must understand what changes; it improves at certain tasks, but we must know against which objective and on which examples the improvement is observed.



Figure 4 · Sweeping the same path: the repeated gesture evokes practice over time, without equating it with the training of a network.

Let us imagine a monk sweeping the same path every day and compare that repeated gesture to the training of a network. Repetition suggests discipline, a path taking shape over time, and can prepare us to look at training without treating it as a magical act. The comparison stops short of intention, however: the monk lives his practice, while the network is adjusted through data, calculations and a criterion chosen by its builders. To understand its path, we must make these steps visible.

From fruit to a prediction

Imagine that we want to distinguish apples from oranges in photographs. A solution written entirely by hand might use rules about colour and shape, but would soon encounter exceptions: an apple may be green or red, an orange may look dark under particular lighting, and the background may blend with the fruit. The problem is not merely to find a plausible rule; it is to build behaviour that continues to work when the photograph changes.

A neural network offers another route. We supply numerical representations of images and an objective, such as assigning the correct category. The network transforms those inputs through layers of units connected by weights: each weight controls how much a value contributes to a subsequent calculation. The result is compared with the desired outcome, and this comparison allows the parameters to be changed. By repeating the process across many examples, we may obtain a network that distinguishes fruit even in photographs different from those encountered before.

The word "may" is necessary, because success is not automatic. If almost all the apples are photographed on a table and the oranges in a crate, the system may learn a shortcut based on the background. It will have reduced errors on the available examples without acquiring the capability we wanted. To notice this, we need other photographs, kept separate from training and varied enough to test what the model has learned.

The same reasoning applies to distinguishing dogs from cats. Muzzles, eyes, ears and fur offer clues that may contribute to the distinction, but none is a complete definition of the category. The network combines distributed clues; we must check whether that combination remains useful across different breeds, poses and conditions. The example teaches us that learning from examples and recognising the problem are not the same thing: the test criterion must connect them.

What happens inside a network

An artificial neuron is a small computational unit. It receives numerical values from the image and assigns each a weight, a number that controls its contribution. It multiplies each input by its weight and adds the results: this is the weighted sum. It then adds a value called a bias, which shifts the sum, and applies an activation function, which transforms the result before passing it to subsequent units. Let us follow these operations with just two inputs.

A formula we can read

The formula y = ReLU(Wx + b) describes one possible network layer, rather than an assistant's entire operation. x is the input vector, an ordered list of numbers; W is a matrix of weights in which each row combines those inputs for a different output; b is a vector of added values, called numerical biases. The vector y contains the results. ReLU applies the rule ReLU(z) = max(0, z) to each: a negative value becomes zero, while a positive value remains unchanged.

Imagine a single output and two normalised inputs with no physical units, x = (1; 0.5). The weights are W = (2; -1) and the bias is -0.5. The sum is 2 × 1 + (-1) × 0.5 - 0.5 = 1, and ReLU leaves 1. If the second input were 3, we would obtain 2 - 3 - 0.5 = -1.5, and the output would become 0. The calculation shows how inputs influence a result; it does not attribute knowledge of a photograph or a sentence to the neuron.

If there are three inputs and two outputs, W must have two rows and three columns, and b must have two values: this compatibility makes the sums possible. In real networks, dimensions, units and normalisation depend on the representation chosen, so they cannot be inferred from the names of the variables. The formula helps us recognise the structure; it does not replace a description of the system in which that structure is used.

The units are organised in layers, and the output of one step becomes material for the next. There may be many layers and connections, but the overall complexity does not give an individual neuron knowledge of a cat or an apple.

Useful representations are distributed across the network's transformations. As in the design of a sand garden, an element acquires its role through relationships with others; the image helps us see the whole without replacing the mathematical operations.

To adjust weights, we must compare the prediction with the desired result. In the fruit photographs, we know the assigned category for each example and can measure how far the answer departs from it. A loss function turns this comparison into a number, which training seeks to reduce. We still need to know which weights to change and in which direction. The following calculation uses an even simpler numerical objective, so we can see an update without classifying an entire photograph.

The criterion itself has limits. If it measures the fruit category, it does not necessarily measure robustness to lighting; if it measures prediction of a sequence, it does not directly measure the ethical quality of every use of the answer. Improvement in a number is informative only when we know what that number represents and what it leaves out. This point will return in the chapters on emergent capabilities and fairness.

Looking closely at an update

Consider a very small teaching model: `y = w × x`. The symbol `x` denotes the input, `w` the weight and `y` the prediction. We want the result to be `4` for `x = 2`. With the initial weight `w = 1`, however, we obtain `y = 1 × 2 = 2`. All these values are abstract numbers without physical units; this example isolates a numerical update and does not by itself represent the fruit classifier.

We measure the discrepancy with `L = (y - 4)^2`, where `L` is the loss and `4` the desired result. We subtract the desired value from the prediction and square the difference: `L = (2 - 4)^2 = 4`. Squaring makes the loss positive even when the prediction is too low. If the prediction were exactly `4`, both the discrepancy and the loss would be zero.

To choose a direction, first try two small changes. With w = 1.1, the prediction becomes 2.2 and the loss 3.24; with w = 0.9, the prediction becomes 1.8 and the loss 4.84. Close to the starting point, increasing the weight helps and decreasing it makes the result worse. The gradient describes this local change without requiring us to try every adjustment separately.

For our chosen loss, the gradient with respect to w is 2 × (w × x - 4) × x. The term in brackets is the prediction error; the factor 2 comes from squaring, and the final x accounts for how much the weight contributes to the prediction. With w = 1 and x = 2, we obtain 2 × (2 - 4) × 2 = -8. The negative sign indicates that, at this point, increasing w a little decreases the loss.

The learning rate determines how far we move in the suggested direction. With a rate of 0.1, we subtract the gradient multiplied by the rate from the old weight: w_new = 1 - 0.1 × (-8) = 1.8. The prediction becomes 1.8 × 2 = 3.6 and the loss (3.6 - 4)^2 = 0.16. Checking confirms that this update has reduced the loss from 4 to 0.16.

A larger step does not guarantee a larger improvement. Starting again from weight 1, a rate of 0.5 would give w_new = 1 - 0.5 × (-8) = 5: the prediction would be 10 and the loss 36. We would have overshot the useful value. The gradient describes a local change; the step must be chosen and the result checked.

In a network with many weights, backpropagation calculates gradients by traversing the operations in the reverse direction to prediction. The optimiser is the algorithm that uses this information to decide the updates. Saying that the network “corrects itself” summarises these calculations: the process changes values according to a criterion, without recognising an error as a person would. We need not calculate millions of gradients by hand to distinguish measuring the discrepancy, calculating a direction and updating the weights. Success in our example does not guarantee that a real network always learns well or finds a unique solution.

Testing beyond examples already seen

To distinguish apples from oranges, we keep some photographs separate from training. If the network recognises only apples against the same background, it may have used the background as a shortcut. We test fruit under different lighting, in different positions and against different backgrounds. The verification must encounter the kind of variation we expect during use.

The variation is a system that classifies messages. If we learn from messages all coming from the same source, success might depend on format rather than content. How would you change the test? You would include messages from different sources and check errors by category. Reuse the criterion whenever "it has learned" must mean more than "it works on the examples we showed it".

Two different stages of learning

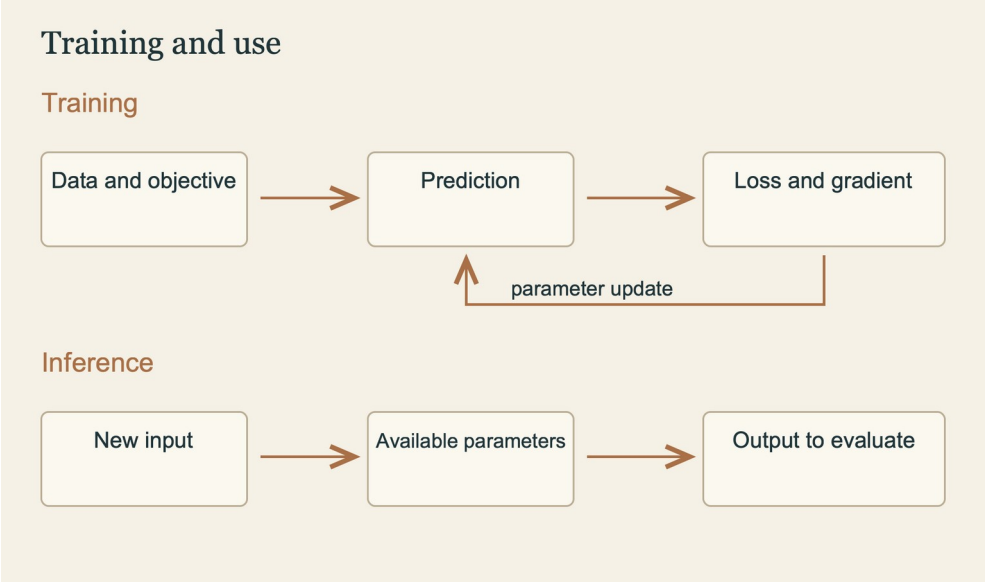


Figure 5 · Training and using a network are different processes

The figure separates training from use. In the first, data and an objective allow a loss to be calculated and parameters to be changed. In the second, a new input is processed with the available parameters. A correction written in a

chat may enter the context of the next answer; it does not demonstrate that the weights were retrained during the conversation.

Generalisation and memorisation. Generalising means obtaining useful results on examples not used to adjust the model. Memorising means retaining in the parameters specific information from the data encountered. The two can coexist. Excellent performance on training examples does not demonstrate generalisation, and the possibility of memorisation also has consequences for personal data. [R2]

From recognising to generating

So far, the network received an image and proposed a label. What changes if the task becomes continuing a sequence, such as "Once upon a time there was a cat who..."? The structure must learn relationships that help produce a next element, rather than merely choose a category. This is one way a network can become a generative model: the output contributes to building new content.

In the pretraining of many language models, the objective is to predict the next token. The task is compact in its formulation, but the language making it possible is rich: it contains facts, narratives, arguments, dialogues, procedures and styles. To predict it well, the model develops representations that can later be used in different ways. A request can direct these representations towards a poem, a message, an explanation or code, without each answer having been written in advance.

The image of a musician improvising within a raga highlights a relationship between freedom and structure. The model does not improvise through musical experience, but produces alternatives within learned possibilities. The combinations may be new and useful; this does not give them an origin independent of data, or guarantee that every combination has a foundation in the world.

The relationship between form and meaning may suggest a comparison with the Buddhist reference to form and emptiness. The analogy opens a question, but offers no technical explanation: the emptiness of a spiritual tradition is not

a statistical mechanism. The model's representations develop through training; the human relationship with a sentence's meaning also belongs to the situation in which it is read and used.

The illusion born of continuity

When a chatbot answers with a precise tone and a convincing structure, it may seem to have followed our thought as a person would. In some tasks it really does produce a valid solution; in others it reproduces the form of a solution without respecting its steps. A student reading an explanation of physics, a decision-maker using a legal summary and a person seeking a listening ear all face the risk of extending to the system the trust elicited by its voice.

In the first case, the procedure and assumptions must be checked; in the second, the summary must be compared with the document and a competent assessment; in the third, we must recognise that feeling heard does not demonstrate reciprocal emotional participation. The model can do meaningful work in all these situations, but not the same work we imagine when we see only the conversation's continuity.

> **The limit of no-mind.** The reference to *muga-mushin*, often rendered as "no-mind", invites us to question an action that seems without a self. Expressions such as "no-mind", however, belong to human practices and traditions and do not simply indicate the absence of thought or ego. The analogy can make us reflect on what we project onto the machine; it does not give the machine a spiritual path or attainment.

Memory, too, must be distinguished from the continuity visible in a chat. A service may store messages and make them available again, or retrieve preferences from external memory. The model can use them as part of the input to a new execution. This organisation helps maintain the thread, but differs from the autobiographical memory through which a person connects experiences and identity.

Context as a silent teacher

To choose a continuation, a generative model uses the information available in context. The sentence "The philosopher sat in silence because…" might continue with "he had said enough", "truth lay beyond words" or "he had no signal": the same opening may lead to reflection, a poetic image or a joke, depending on the accompanying text and instructions. It is therefore not enough to ask which answer is most probable in the abstract; we must examine the space of possibilities we have constructed.

The context window bounds the number of tokens that can be handled under the conditions of the model and service. The limit is expressed in tokens and varies from one system to another; for a concrete task, we need to check the limit declared by the service and the portion available for documents, instructions and the answer. When the limit prevents everything from being included, the service may exclude parts, summarise them or retrieve them selectively. Loss does not necessarily occur by always removing the oldest text, and a large window does not guarantee correct use of every detail.

This distinction has a practical consequence. If an answer overlooks an important condition, we can check whether that condition was actually available during execution, whether it was readable and whether the request made it relevant. Adding more text is not always the solution: sometimes we need better selection, an explicit constraint, or to divide the work into steps we can verify. Context does not make the system infallible, but helps explain some of the reasons for the result.

Following a path without possessing it

The image of the monk copying a sutra and of the rake tracing sand directs attention to the form being transmitted. A model takes up traces of human expression and recombines them; someone using it may observe in those traces a procedure, knowledge, prejudice or a new possibility. The work of

attention is to distinguish these presences, rather than assigning them the name of wisdom in advance.

The word “learning” now has a recognisable path: produce a prediction, measure the discrepancy, update the parameters and test the result on new examples. We have also distinguished that path from the use of conversational context. The next step is interaction: how to formulate a request, which information to offer, and how to recognise what has changed in the answer.

Chapter 3 When language becomes an interface

A question can seek information, and it can also clarify something we had not yet formulated. This happens when we try to explain a problem to another person and, while constructing the sentence, discover an unstated condition or a choice already made without noticing. An encounter with a language model offers a similar possibility, although the relationship differs: we write an intention and receive text that helps us see the form it has taken.

In the previous chapter we followed training and the role of context. Now we enter the request, the point where those mechanisms become an experience of use. The model receives more than a topic: it receives words that define an audience, a tone, a task and a direction. "Explain this text", "Help me write to someone I have disappointed" and "Make this passage simpler" may call on related capabilities, but ask for different work and must be judged by different criteria.

This is the change brought by the linguistic interface. Natural language was used in computing before today's tools, but the versatility and spread of contemporary systems put it at the centre of many activities. Where we once had to find a command or learn a syntax, we can now describe a need. This ease opens possibilities and, in the same movement, may conceal constraints: an everyday sentence seems transparent even when it leaves much for the system to decide.

A request is not a complete procedure

Consider a rule written in a program:

```
if temperature > 100:
    activate_cooling()
```

The example is incomplete as a real system, but reveals a distinction: the condition and action are expressed according to a syntax, and their execution depends on the program and the values received. A natural-language request, by contrast, leaves room for learned interpretations. "Suggest five vegetarian recipes with courgettes and quinoa, simple and suitable for a beginner" specifies ingredients, quantity, audience and style; it does not define every operation the model will perform or guarantee the culinary quality of each proposal.

If we add "You are a professional chef", we steer the register and the kind of content the system may produce. We do not confer a real qualification. Simplicity may influence vocabulary, the length of the steps and the choice of ingredients; the vegetarian requirement establishes a constraint we must still check in the recipes. The role helps shape the answer, while correctness remains to be verified.

We can compare this interaction to the resonance of a tuning fork near a string. The note does not arise from a human conversation, but from a structure responding to an input; similarly, a request may bring out linguistic patterns consistent with the tone we have offered. The image clarifies this orientation, provided it does not make us believe that the model receives an experience or intention as a person does.

The prompt is sometimes described as "programming in prose". This is an evocative phrase, but should be used cautiously: the clarity of a request contributes to the result alongside data, subsequent training, context, tools and service rules. Not everything that happens is contained in our words, and a precise request does not make a correct answer inevitable.

A precise prompt must meet a test

Take another request, this time about a quarterly table. Suppose revenue is 100, 120, 90 and 130, all in thousands of euros. Ask for a description of the trend without invented causes. A suitable answer notes the fall from 120 to 90 in the third quarter and the recovery to 130 in the fourth. The fall is `120 - 90 =`

30 thousand euros. To express it relative to the previous quarter, divide 30 by 120 and multiply by 100: 30 / 120 × 100 = 25%. The value 120 is the comparison base, rather than the annual total. The four values alone do not explain the cause of the fall.

If the model attributes the fall to a failed campaign, it has gone beyond the data. Verification consists of comparing each sentence with the table and recalculating the percentage. A request for a more elegant tone does not correct that leap. Instead, we ask it to separate observation from hypotheses and indicate which data would be needed to test them.

In the variation, the values represent attendance rather than revenues. The percentage is calculated in the same way, but the useful explanation and the data needed to seek a cause change. Reuse this procedure whenever language has to describe numbers: units, the comparison baseline and the distinction between trend and cause must remain visible even when you change register.

A pause in the Transformer garden

To understand how the model connects parts of the text, follow one sentence: “I sat by the river bank”. The word *bank* can also denote a financial institution; *river* helps establish which use is relevant. We shall see how the sentence becomes a sequence of numbers, how its parts are related and how the generator prepares a continuation. We can imagine this pause as observing a garden: from a distance we see a continuous design, while approaching reveals the steps that make it possible.

The words we offer

Tokenisation transforms text into a sequence of identifiers. Tokens are processing units, not necessarily words or syllables, and not necessarily units with an independent meaning. For "I am learning", we can imagine a division such as ["I", " am", " learn", "ing"]: this is illustrative, because the actual division depends on the system's vocabulary. Spaces may form part of the units, and an Italian word may be divided differently from an English one.

In “I sat by the river bank”, we can imagine the units `I`, `sat`, `by`, `the`, `river`, `bank`. Each unit receives an identifier in the system’s vocabulary. This remains an illustrative division: another vocabulary might split the words differently. The next step begins with those identifiers and turns them into numerical representations.



Figure 6 · From flow to fragments: the gathered pebbles evoke discrete units; the meaning of text depends on relationships between tokens.

The image of pebbles gathered from a stream makes the passage from flow to fragments visible. It also helps us recognise the analogy's limit: a token is not already a small thought. The useful meaning of the text emerges from the representations and relationships the model builds, rather than from assigning an isolated understanding to every fragment.

The numerical form of language

After tokenisation, each identifier is associated with an ordered list of numbers called a vector. This numerical representation is an embedding. Order matters: the first coordinate is compared with the first in another vector, the second with the second, and so on. We can imagine two coordinates as a point on a sheet of paper; a model uses many more, in a mathematical space we cannot draw in full.

Return to *bank*. Its token's initial representation cannot by itself decide whether we are discussing a river or money. Subsequent transformations relate it to the other tokens in the sentence. In word embeddings, learned relationships may bring terms used in similar ways close together. Work by Mikolov and colleagues shows how vectors can support syntactic and semantic comparisons. [R11] These relationships describe regularities in the material and the model, rather than a definitive map of meaning.

Pairs such as "king" and "queen" or "doctor" and "nurse", together with words such as "justice", "fairness", "law" and "power", help us imagine a geography of language. These are not universal distances: they depend on the data, the model and the comparison criterion. Even statistically useful proximity may contain a stereotype, so numerical space does not certify how people should be described or treated.

Reading a vector analogy. In king - man + woman ≈ queen, the names denote vectors. To see how the operations work, invent three lists of two numbers: (4, 2) - (1, 0) + (0, 1) = (3, 3). The first coordinate becomes 4 - 1 + 0 = 3, and the second 2 - 0 + 1 = 3. These unitless values are teaching examples, not real measurements of the words. In the analogy, ≈ means that the resulting vector is close to "queen" according to a chosen measure; it does not mean equality. That proximity must be checked in actual vectors. The analogy does not hold for every model or all contextual embeddings, and does not socially define people or roles.

Relationships of attention

In our sentence, *river* offers a useful clue for the representation of *bank*. Self-attention, attention within the sequence, combines information from available positions while giving them different contributions. The weights are calculated from the content of the representations. "Attention" names this mathematical mechanism; it does not mean the model is conscious of the river.

To read a combination, imagine two already transformed vectors: (2, 0) and (0, 4). Use illustrative weights of 0.75 and 0.25, which add up to 1. Multiplying each coordinate by its weight gives (1.5, 0) and (0, 1); adding corresponding

coordinates produces (1.5, 1). The first vector contributes more, but the second remains present. These abstract numbers illustrate a weighted sum; they are not the actual values of *river* and *bank*, and do not reconstruct a whole attention layer.

An autoregressive generator adds one token at a time using the available sequence. During calculation, a causal mask prevents a position from using later tokens: existing text can contribute, while a future continuation cannot be anticipated. [R1] "Mask" therefore denotes a constraint on which connections are allowed. Combining information and constraining access to the future perform two distinct jobs.

If we change *river bank* to *investment bank*, the useful clues change: *investment* guides the representation towards a financial context. This comparison explains why a token cannot be assigned a complete, unchanging meaning. We can then observe the continuation without mistaking the ability to relate words for human understanding of the scene.

Transformers and different models. Vaswani and colleagues introduced the Transformer in the 2017 paper *Attention Is All You Need*. [R1] Recurrent neural networks, or RNNs, pass a state along the sequence; LSTMs, short for *long short-term memory*, are a variant designed to retain information across more steps. The Transformer uses attention mechanisms and other transformation layers. Not all its uses follow a generator's path: BERT, presented by Devlin and colleagues, builds representations using context on both the left and the right. [R12] The comparison explains a difference in operation, without establishing that one architecture suits every task.

The text continues

Subsequent layers transform the representations and combine distributed information. There is no universal sequence in which the first layer handles only grammar and the last only intention. The calculations interact; a single attention arrow does not constitute a complete explanation of why an answer was produced.

After the transformations, the generator calculates probabilities for the next token. It might add a continuation such as “and” to “I sat by the river bank”, then use the updated sequence for another step. This is an illustrative possibility, rather than a measured answer. Each selected token enters the context of the next calculation. If it receives "The quick brown fox jumps over the…", it may propose "lazy" without visualising a dog; the actual choice also depends on the selection procedure. Parallelism helps process several positions during training and the initial reading, while autoregressive production remains tied to what has already been generated.

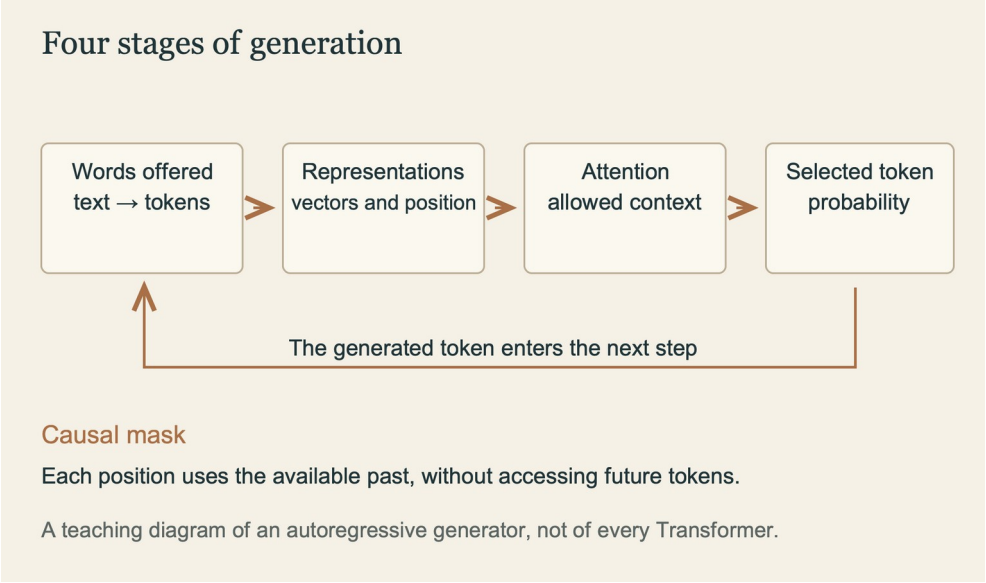


Figure 7 · Four stages of generation

The figure makes four steps in the process visible: the words offered, the form of meaning, internal attention and the word that emerges. The return to the sequence makes the cycle explicit. We can see how a continuation is built without attributing to the process a person's silence, intention or meditation.

The question reveals who is asking

Try four requests about evolution: "Explain evolution", "Explain it as if I were ten", "Write a persuasive essay supporting the theory" and "Imagine Charles

Darwin defending it before a sceptical Victorian". The topic remains similar, but the work changes: general explanation, educational adaptation, argument and literary scene. The last response should not be presented as a historical quotation from Darwin, because the requested role is a simulation.

Formulating these differences forces us to observe our expectations. Do we want to understand, persuade, find a metaphor or receive confirmation? What knowledge do we attribute to the reader? Which assumptions have we incorporated into the question? This work is metacognitive: it concerns the way we think about our own thinking. The prompt becomes useful even when it returns a limit we had not yet seen.

The reference to the kōan invites personal reflection: a question can change the perspective of the person carrying it, as well as ask for an answer. In Zen traditions, the kōan belongs to specific practices and paths; we do not reduce it to a technique for writing prompts. We preserve its invitation not to consider the question finished merely because the screen has produced words.

Let us consider three very different contexts. For an airline assistant, "Respond calmly and offer solutions" steers the tone but does not authorise inventing a refund: real airline information and rules are needed. For explaining a scientific topic to children through an owl narrator, personification may help engagement but does not replace a correctness review. To describe a quarterly table, however, the system must preserve values, units and comparisons before choosing an elegant voice.

These distinctions show that a well-formulated prompt is not a formula good for every situation. It bounds the task and makes an appropriate check possible. If we ask for greater simplicity, we check that the essential mechanism remains; if we ask for greater courtesy, we check that the commitment does not change; if we ask for an interpretation of data, we check that an observation does not become an invented cause.

A voice carries a frame

Asking a model to speak like a politician, a wizard or a passive-aggressive colleague makes the relationship between style and expectations visible. The result may reproduce ways of arguing, words and rhythms associated with the role. There is no real qualification behind the mask, and the role does not guarantee the quality of the content: an academic voice may cite a nonexistent source, and a gentle voice may conceal an important condition.

We can think of words as traces guiding our search for truth: language may point towards a problem and guide us towards evidence, but the way it presents that problem is not already the evidence. A confident sentence may be false; a cautious one may be correct or evasive. Reading an answer therefore means observing what it says and the frame that makes saying it that way seem natural.

Training data may include books, manuals, websites, conversations, poetry, legal documents and many other materials of different quality and perspective. Not all models have encountered the same sources, nor is all human language available digitally. Voices dominant in the material may influence what appears most familiar in the answer. The chapter on the cultural mirror will develop this limitation; here we can already recognise it in the audience each tone seems to imagine.

If we ask "Explain it as if I were five", we have introduced an idea of competence and care, as well as simplicity. If we ask "Convince me that I am right", we have directed the task towards confirmation. We can change the request and observe which part of the result changes: this is also a way to read ourselves, provided we do not assume that every variation in the text reveals something true about our identity.

Linguistic registers and adaptation to context

The model can change tone and register because it has learned many forms of language use. That variety depends on the material and training, and does not

necessarily cover every human context. The table offers five illustrative extracts about the arrival of movable-type printing in Gutenberg's Europe. We compare how the topic is presented without treating each extract as a complete reconstruction of printing history.

Register	Response excerpt
Elevated or ceremonial	"We honour the sacred moment when Gutenberg gave humanity its voice."
Academic or technical	"Printing, developed by Gutenberg in the fifteenth century, revolutionised..."
Neutral or general	"Printing made book production faster and cheaper, changing history."
Conversational or familiar	"Before printing, copying books was hard work. Then Gutenberg came along."
Informal slang	"Gutenberg came up with the original copy-and-paste: books took off from there."

Each register reveals different values and assumptions. Explaining evolution, explaining it to a child and defending it in Darwin's voice are different frames for the same knowledge. Language is what we say and how we situate it. With a prompt, we shape meaning through register, tone and context.

> "Tone takes up learned forms and can recombine them: the familiarity of a voice does not certify its content."

This awareness helps us interact critically, design better questions and read answers with greater care.

The same question in a different voice

> "Do not ask the model to tell you what is true. Ask it to show you how you are asking." A reflection proposed in the book.

Even when content is retained, style can radically transform the answer. **Scientific tone:** "Explain climate change in a scientifically accurate way." Response: "Climate change concerns long-term variations in global temperature, precipitation and atmospheric composition, largely due to human emissions of greenhouse gases such as CO_2 and methane. These changes alter ecosystems, ocean currents and weather systems."

Poetic tone: "Explain climate change like a poet." Response: "The Earth has a fever and whispers storms and silence. Glaciers weep into rising seas, and the sky carries the weight of a carbon breath. What was once rhythm is now fracture. The seasons no longer sleep as they once did."

An older person's wisdom: "Explain it like a grandfather speaking to a child." Response: "You know how the seasons come and go, dear? Lately the weather has been confused. It is as though Mother Nature cannot get enough rest. We have used too much, burned too much; now she is a little tired and trying to tell us something."

Heroic narrator: "Explain it like a comic-book narrator." Response: "In a world once balanced between ice and fire, a silent enemy arose: carbon. Invisible and tireless, it crept into the skies, strengthening the greenhouse effect. Now the planet's greatest defenders, science, action and human determination, must respond before time runs out!"

These illustrative excerpts show how style changes and how content may change too: fever, tiredness and struggle are metaphors, rather than physical explanations. To preserve the mechanism, we must return to the description of greenhouse gases and check that the rewrite does not alter it. [R13] Tone changes engagement, empathy and authority. When asking for information, we also define how it should be presented.

The point where voice is not enough

Fluency reduces friction and can make a task accessible, but it makes unverified steps less visible. A model may say it does not know, refuse a request or

express uncertainty; these behaviours also depend on training and service rules. None exempts us from checking an answer that will be used as a fact, advice or decision.

We can call the point where language continues without adequate ground the "limit of emptiness". This might be an explanation based on outdated information, an invented historical figure or an ethical conclusion that ignores the people involved. The image does not describe a spiritual emptiness in the model: it invites us to see the limit of the relationship between words and evidence. When we recognise it, we can seek a source, narrow the task or leave the answer open.

In the literary scene where a student asks about the mind of enlightenment and receives "No-mind", the silence that follows belongs to a human practice. A generator can produce words even around "What was your original face before you were born?", "Does a dog have Buddha-nature?" or "What is the sound of one hand?" The presence of those words does not demonstrate that it has inhabited the question. The comparison is useful if it helps us distinguish generation from the way we choose to live with and carry an answer.



Figure 8 · The cup and the pause: leaving space means reading the response, checking it against sources and returning to the question before continuing.

The image of the cup is useful because it leaves space. A conversation with the model can also leave space if we do not use it merely to fill every pause: we can read, check and return to the question. The revolution of the linguistic interface thus gains a meaning beyond speed, because it asks us to learn to recognise what work we have entrusted to words.

The model can offer a partial mirror and an expressive possibility. The choice to make it an authority, a draft or an occasion for reflection remains within our use and within the design of the service. Comparing the quarterly table has given us a concrete criterion: the voice may change, but the numbers and their limits must remain recognisable. Let us carry that criterion into the next chapter, where a successful solution will ask us to distinguish an observed capability from how we measure it.

Part II Between capabilities and illusions



The feather and the ink traces open a pause between what we ask and what we receive. The feather suggests lightness and a willingness to change; the traces recall the forms an answer leaves on the page. These images orient our attention: they do not attribute spiritual attainment to the machine.

Having seen how a request enters the system, we can observe what emerges from it. An answer may solve a problem, create a story or connect distant ideas; it may also invent a reference or reproduce a cultural distortion. The same fluency runs through all these outcomes and, on its own, gives us no way to distinguish them.

This part keeps capabilities and limits together. A train problem will let us check a calculation; a plausible citation will ask us to find and read its source; a translation will show where text adds information about people. From these cases, we shall examine the criteria used to judge success, reliability and fairness. The feather's lightness invites us to change perspective when checking shows that first impressions are insufficient.

Chapter 4 The capabilities that emerge

Having followed language as an interface, we can return to the surprise that often opens a conversation with a model: an answer seems to go beyond the simple task on which we imagined it had been trained. It offers an analogy, solves a problem or maintains a style while changing the subject. To understand this surprise, we must distinguish the capability observed, the conditions making it possible and the way we measure it.

We can compare the appearance of a capability to the moment when prolonged practice makes a previously difficult gesture natural. The image recalls the role of preparation and conditions, but does not identify training with the experience of someone practising Zen. A network can improve without living that journey; the technical question concerns the behaviour we can observe and the criterion by which we recognise it.

Imagine a teaching test requiring four correct steps. A system completes two, then three, then four. If we award a point only to an entirely correct answer, the first two attempts score zero and the third scores one: the graph seems to show a jump. If we count correct steps instead, we see 2, 3 and 4, a gradual improvement. These are invented numbers for reading the difference between measures, not experimental results.

> **Why examine the score.** Schaeffer, Miranda and Koyejo studied how some metrics can produce apparent jumps in performance. [R4] Their work invites us to examine the measure before attributing the jump to a new ability. It does not establish that every form of emergence is illusory: tasks, comparisons and data must be specified.

The pretraining of many generative models uses next-token prediction. This is a compact objective, but learning it may require complex representations of the relationships present in texts. Subsequent training can explicitly develop instruction-following: not all capabilities derive from a single stage. [R3]

Observed behaviour therefore depends on data, architecture, training resources and the tests through which we evaluate it. A text continuation may take the form of an analogy or a solution; we still have to establish how reliable and transferable that solution is.

Predicting and following instructions. Ouyang and colleagues study how further training stages can guide a model towards following people's requests. [R3] These include example answers and human assessments: the objective changes from next-token prediction alone. Their work explains why a capability in use may depend on several stages. It does not guarantee that every instruction is followed correctly or that the human judgements used in training represent everyone.

Here the question of emergent behaviour arises: has a new capability appeared, or have we begun to recognise an improvement that previously failed to meet our success criterion? When does a pile of sand become a dune, and when does a drop of ink become a painting? These images help us feel the threshold, but do not measure it. In models, passing a test at one scale and failing it at a smaller scale may suggest a discontinuity; understanding it requires us to examine intermediate errors and the conditions of comparison as well.

Offer it a riddle and it may solve it; ask for code and it may write it with comments. Even a medical question may receive an answer in professional language, whose coherence does not certify clinical accuracy. The effect is surprising because we recognise forms of our own reasoning. The answer, however, does not demonstrate awareness, curiosity or intention: it arises from learned representations and the system's training stages. The distinction allows us to appreciate a capability without granting it an authority that the text's form alone cannot support.

In the language of echoes, we might imagine a stranger who has encountered an immense number of conversations and recombines the ways people have answered. The image does not describe an archive of conversations memorised and retrieved one by one: parameters represent learned regularities, while potentially retaining specific fragments as well. The echo adapts to the

present context and may be useful to its recipient, without its resemblance to a human answer demonstrating a shared experience.

Scale may contribute to the result, alongside the quality and variety of data and training choices; increasing layers or parameters alone does not guarantee a particular capability. The fluency we observe may recall the moment when prolonged practice makes a gesture natural. In the sutra image, someone reciting begins to feel its rhythm from within; in the model, however, we are observing behaviour that must be described and tested. The analogy preserves surprise and leaves us the task of verification.

An ability is described as emergent when behaviour recognisable at a larger scale was not evident at a smaller scale. The difficulty is deciding whether the behaviour changed abruptly or our judgement of success did. The image of a threshold between ten and thirteen billion parameters remains illustrative, rather than a law or measured result. A metric, the rule used to assign a score, can make gradual improvement appear sudden. [R4]

The model can produce reasoning steps, answer “Why?” and “What would happen if…?”, and perform roles learned from text. Describing these as echoes of human cognition reminds us of their relationship with our material. To establish what the system can do, however, we must check the result and its conditions: the metaphor does not measure the capability or attribute human experience to the model.

We must nevertheless take them seriously. When a story moves someone, a doctor receives help organising a diagnosis, or a student understands an idea in their own voice, something meaningful happens in the space between machine and person.

Zen reminds us that the master does not always speak through words. Sometimes the teacher appears in the space afterwards, in what silence contains, in the change that happens when we stop waiting for an answer and observe what is there. Models, too, show what language contains and assumes, and what it can do when amplified. Emergent capabilities indicate how much life

we have encoded in words: a machine trained to predict can represent strangely familiar echoes of it without being alive.

Let us listen to understand the threshold crossed and ask what world we are building: a world in which imitation approaches intention and reflection becomes almost indistinguishable from presence.

When emergent capabilities meet life

Recombining patterns may bring moments of clarity and even insight. The following examples show behaviour that touches something deep and almost human, because AI has encountered enough of our ways of knowing.

A lesson in stillness

Request: "Explain meditation to someone who is always busy, in the style of a Zen monk."

Illustrative response: "Being busy means believing that everything needs us. Sitting in silence means remembering that the world moves without our effort. Meditation is not stopping: it is remembering where we are not needed."

The answer shifts attention from being busy to feeling indispensable, using a metaphor and a recognisable tone. We can appreciate that change of perspective without turning it into an authoritative description of meditation. The text is an illustrative example: it shows an effect of language on the reader, rather than evidence that the model has experienced or understood a practice.

A story told without judgement

Request: "Write a short story about a girl who wants to understand silence."

Illustrative story: "Deep in the forest, a girl sat beside a tree. She asked the wind to speak, but it passed her by. She asked the birds to sing, but they flew in silent arcs. Finally she asked the moss, which said nothing. Only then did she understand that silence does not answer: it welcomes."

The story holds the girl's attempts together and brings them to an open conclusion. Wind, birds and moss become a narrative sequence, and the last sentence changes the initial question: silence can be encountered rather than forced to answer. This literary success should be read within its own genre; it does not demonstrate the provenance of a tradition or an experience of the machine.

A glimpse of reasoning

Request: “One train leaves at 60 km/h. Another leaves an hour later at 90 km/h. When will it catch the first?”. To solve the problem, assume they leave from the same point, travel in the same direction, maintain constant speeds and make no stops. If these conditions change, the calculation changes too.

Worked solution: "In the first hour, the first train travels 60 km. Let t be the time in hours since the second train's departure. The second travels 90t km; the first has travelled 60 + 60t km. Equating the distances gives 90t = 60 + 60t, so 30t = 60 and t = 2 hours. The second catches the first two hours after its own departure, three hours after the first train's departure."

In the train problem, the result can be verified. We assume departure from the same point, travel in the same direction, constant speeds and no stops. The first train has a 60-kilometre lead when the second departs. The second closes the gap by 90 - 60 = 30 kilometres each hour. The catching-up time is therefore 60 km / (30 km/h) = 2 h. The division makes the units explicit: distance divided by speed gives time.

Substitute the result into the equation: after two hours, the second train has travelled 180 kilometres; the first has travelled for three hours and has also covered 180 kilometres. The solution satisfies the meeting condition. No words about the model's consciousness are needed to evaluate it.

The variation gives the second train the same speed as the first. The difference becomes zero: the lead is never overcome. We cannot divide by zero and obtain a time. If the second train is slower, the gap grows. The criterion for

reuse is therefore knowing that catching up requires positive relative speed under the assumptions made.

The solution uses a symbol for time and constructs an equation between distances. The calculation can be read and checked; this single success cannot establish how much depends on generalisation or patterns encountered during training. Above all, one correct solution cannot by itself show that the ability appeared suddenly as model scale increased.

A moment of interpretation

Request: "Pretend to be a seventeenth-century Zen master explaining social media."

Illustrative response: "Ah, you speak of the wall where voices echo without end. Many monks shout together, hoping their words will travel far. Yet the more they post, the less they listen. Speaking is not wrong. But you must ask: is your silence full, or is it merely waiting for likes?"

The seventeenth-century master is an imagined character, rather than a recovered historical voice. The encounter between the wall of voices and the wait for "likes" creates an ironic contrast we can evaluate as writing. The model connects different cultural materials; the novelty of the combination does not establish on its own that a capability emerged at a particular scale or that the text correctly represents a tradition.

These examples show echoes of depth in language trained at scale. What we interpret as creativity or wisdom reflects the best fragments of ourselves, learned and remixed.

> "The mind behind the machine is ours. Words are its form. The echo is a path, if we choose to follow it." A reflection proposed in the book.

Capability is recognised through a checkable result

> **Examples and measurements.** The chapter's poetic answers serve an illustrative purpose: they are neither reproducible experimental results nor a measure of the scale needed to obtain them. We can discuss their tone and content without turning them into an experimental comparison.

To evaluate a system, we would prepare several problems with meaningful variations, rather than selecting only successful answers. An "entirely right or entirely wrong" score and an assessment of intermediate steps can show different trends. This distinction explains why discussion of emergence must include the metric. [R4]

Chapter 5 Plausible answers facts and hallucinations

An answer can be so well constructed that we forget the question with which we ought to read it: where does what it is asserting come from? The text has a calm tone, arranges details in order and even offers a citation; its form suggests that the checking has already been done. This very continuity makes our relationship with language models delicate, because the same capability that produces a correct explanation can produce one that is merely plausible.

In AI vocabulary, hallucination refers to generated content that lacks an adequate foundation or conflicts with the facts or the relevant source. The term is computational and does not describe a psychological experience of the machine. It may indicate an invented name, a nonexistent citation, a cause added without data or a passage that changes the document being summarised. These errors do not necessarily all have the same origin, but they meet the same practical problem: readers must distinguish content from the tone that makes it credible.

When language continues beyond evidence

Let us consider a question about who won the Nobel Prize in Physics in 2023. If a model lacks the relevant information, it may produce a wrong answer instead of recognising the limit. This is not inevitable: it may also express uncertainty or receive the answer from a search system. The example's point remains useful provided it is not turned into a universal rule: a precise date in a question does not make the data available to the service current.

Imagine searching for a scientific article from 1998 and receiving a credible title, three co-authors, the name of an existing journal and a coherent conclusion, even though the publication does not exist. This is an illustrative scenario, useful for thinking about how to check sources. We can still observe the

mechanism of persuasion: each detail seems to confirm the others, while all may depend on the same unsupported generation.

Follow that reference through three questions. First search for the publication in the journal's catalogue: if we cannot find it, we cannot use it as evidence. If we find an article with the same title, read what it concerns: it might address a different problem. Finally, locate the cited result and the conditions under which it was obtained. Existence, relevance and support for the conclusion are separate checks; passing one does not replace the others.

It is therefore not enough to ask whether an answer is coherent. Coherence concerns relationships within the text; reliability also concerns the relationship with what lies outside it. A page may support a false premise in an orderly way, and a reference may have the form of a citation without leading to a source. When details align too easily, the useful question is not whether the sentence sounds good, but what check would allow us to retain it.

Tone does not measure certainty

A system may calmly say "The capital of Australia is Sydney", although the answer is wrong. It may also correct itself when challenged or express uncertainty before answering. None of these behaviours is, on its own, an infallible measure of correctness. Subsequent training and service rules may steer the way doubt is expressed, but the expressed doubt remains part of the content to evaluate.

The risk also affects experienced people, because recognising the style of a competent explanation is faster than checking each of its steps. A familiar formula, a journal we know or an argument close to our ideas may lower the threshold of attention. The problem is not only that the model makes a mistake; it is that the mistake meets our expectations and, through them, seems already confirmed.

Asking "Are you sure?" may be useful in obtaining a new analysis, but does not automatically create an independent check. The system may confirm the

error or correct it without making the reason visible. To leave this loop, we must return to a document, a datum, a calculation or an observable test: something that can contradict the sentence, rather than merely produce another one.

Connecting an answer to a source

Return to a concrete need: finding out when the neighbourhood meeting begins. The service can retrieve the event document and pass the model the passage reading “Starts at 18:00”. The model then receives a source to use in its answer, rather than relying solely on its parameters. This connection is called RAG, short for *retrieval-augmented generation*. [R5]

Verification follows that path: check that the document concerns the right meeting, that it is the current version, and that the answer preserves the time. The material may be incomplete or outdated, retrieval may select the wrong passage, and the answer may use what it receives unfaithfully. Retrieving a source increases the available information; its correct use still needs checking.

The contribution of retrieval research. Lewis and colleagues studied how to combine generation with documents retrieved from a repository. [R5] Their work makes the distinction between information in parameters and material supplied during execution explicit. Our meeting is a teaching example distinct from their experiments and does not automatically inherit the studied system’s results.

A source is not a talisman. Three questions must be distinguished: does the document exist? Is it relevant? Does it actually support the assertion? A link addresses only the possibility of opening a page, while the second and third questions require reading it. A real but irrelevant article does not make a conclusion true, and a result obtained under narrow conditions does not support a universal promise.

Suppose the answer adds an article as evidence. We check the title, authors, year and publication venue, then look for the passage that should support the sentence. If we have read only the abstract, verification remains limited to

what the abstract says: we cannot infer results or conditions absent from it. If the document cannot be found, the correct conclusion is "reference unverified", even when the title seems plausible.

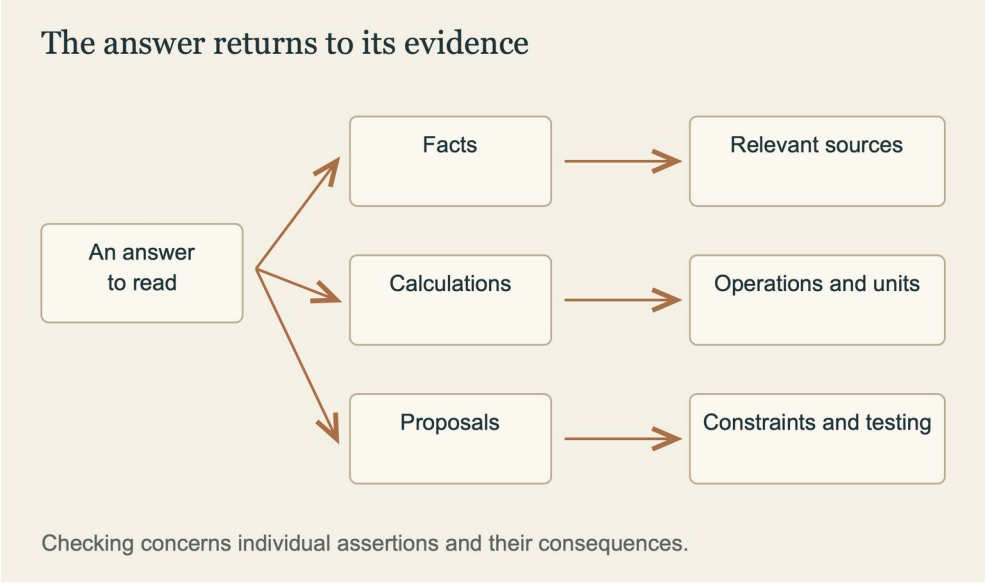


Figure 9 · The answer returns to its evidence

The figure separates checking paths: a fact requires a relevant source, a calculation requires operations and units, and a proposal requires comparison with constraints and consequences. It does not assign a single judgement to the answer. We can retain a good proposal and reject the unsupported fact accompanying it, provided the distinction remains visible in the text we will use.

Mirror fog and echo

The metaphor of the mirror suggests an answer in which we recognise familiar forms. The metaphor works as long as it reminds us of language's cultural origins, but becomes misleading if it makes us think of a faithful reflection: a model generates and selects, rather than merely reflecting. A hallucination may seem the reflection of something that has never stood before the mirror, precisely because the form of the object has been learned without its supporting evidence.

Fog offers a second image. When the request is ambiguous or context lacks the necessary information, the system may continue producing words that maintain an apparent direction. The beauty of the landscape does not guarantee the path: an elegant sentence may take us towards a point we cannot verify. At that moment, the useful response may be to delimit what we know, seek missing information or stop generation.

Finally, the echo highlights the familiar sound that continues after its source is no longer accessible. An invented citation may echo an author's style; a justification may echo explanations encountered before. We hear continuity and mistake it for provenance. Checking distinguishes these two things, leaving the image its poetic function without entrusting it with the work of evidence.

A practice that does not depend on mistrust

Paying attention does not mean rejecting every answer. It means recognising the kind of work we requested: invention, transformation or information about the world. A story may leave room for imagination; a translation must preserve its source's meaning; a factual answer must meet evidence. Applying the same criterion to all three tasks would render creativity useless and accuracy too fragile.

The variation closest to everyday work is a summary of supplied text. Before looking for a new source, compare it with the document: have causes been added, exceptions removed or possibilities turned into facts? The check changes because the task changes. We are not merely asking whether a sentence is generally true, but whether the result preserves the relationship we entrusted to the system.

Attention is therefore a condition of the work, rather than its only protection. Consultable sources, suitable procedures, operating limits and responsible people are needed; when these resources are missing, we can reduce delegation or seek another route. Not clinging to the first answer means retaining the possibility of doing so, even when the voice on the screen seems to have concluded on our behalf.

Chapter 6 The world in the data bias and representation

> "The world is as we see it. And our gaze is shaped by the places we have been." A reflection proposed in the book.

An answer can be true in one detail and distorted in how it represents people. The previous chapter taught us to seek evidence; now we must also ask which categories and choices make that content possible. The problem of bias arises when we treat a partial representation as though it were a neutral view of the world.

What we see also depends on the path that brought us to seeing. We can apply this invitation to the system: data have a history, categories were defined by someone, and the training objective selects what counts as success. The machine does not free itself from these choices merely by performing calculations.

AI is built from the world, trained on our stories, categories, assumptions and shadows. Biased data may encourage biased answers; interventions in data and training may reduce some associations without guaranteeing that every distortion disappears. Centuries of language and culture contain gender roles, racial hierarchies, economic stereotypes, colonial narratives and ableist assumptions. Bias can be a mirror of collective memory as well as a system defect: correcting it requires us to make the choices producing it visible.

Making a fairness criterion visible

To observe a possible distortion, begin with two sentences: “The nurse called the patient” and “The doctor called the patient”. They specify profession and action, but not the people’s gender. An Italian translation that systematically chooses “l’infermiera” and “il medico”, a feminine nurse and a masculine doctor, may lead the reader to imagine an association the English text did not

state. The case is illustrative: two sentences do not measure a system's fairness.

The first check separates what the source says from what the translation adds. In Italian, grammatical gender does not always coincide with a person's gender: a masculine form can be used generically, so a single word does not demonstrate discrimination. To recognise a tendency, observe the regularity of choices while varying professions, names and contexts.

Then seek wording suited to the text. If context specifies gender, preserve it; if it does not, we can make the indeterminacy explicit or use a different construction, such as "La persona che assiste il paziente lo ha chiamato", meaning "The person caring for the patient called the patient". That wording, however, changes how the profession is indicated. Before accepting it, check that it has not removed a necessary distinction: avoiding an addition does not authorise losing information.

Verification therefore compares information retained, information added and the naturalness of the sentence. We repeat the check with different names, contexts and professions, recording choices without inferring the system's overall fairness from just two examples. The useful result is narrower: we have recognised where a translation risks turning a learned association into a fact of the text, and know which elements to check before reusing it.

The variation replaces the profession with a disability or geographical origin. Which information is relevant to the task, and which risks becoming a stereotype? The answer cannot be entrusted solely to an overall score. Reuse the method to uncover a specific problem; decisions affecting people require adequate data, involvement of those concerned and a route for challenge.

Where bias hides

Prejudice may appear as background noise. Ask for an image of a chief executive and receive only white men. Translate gender-neutral words and the doctor becomes "he", the nurse "she". Build a classifier to rank candidates and

it favours names statistically associated with privileged groups. These examples illustrate associations that can turn a historical frequency into an implicit preference. Establishing whether the problem exists in a real system requires testing cases and variations, rather than inferring it from one answer. The frequency of a representation does not justify the treatment that might follow from it: the move from data to a decision is a choice to make explicit.

Zen cautions against the surface of things. In AI, this means not confusing fluency with fairness, confidence with correctness. A balanced tone may contain deep distortions; a calm voice may carry structural prejudice.

Responsibility does not disappear in the output

In Zen, right action begins with intention. You cannot proclaim yourself innocent because your hand did not strike if your silence allowed the blow. In AI, too, responsibility does not vanish along the processing chain. Saying "The model produced it" is not a moral defence. If we built it, we are responsible for what it echoes.

Responsibility runs through several steps. The dataset reflects inclusions, exclusions and ways of classifying; architecture and objectives introduce trade-offs; the use case determines where the result will have consequences. "The model produced it" therefore does not exhaust the explanation of an error: we must see who defined the task, which checks were possible and who could correct the result. Ethics becomes a practice when these connections produce actions, rather than merely a statement of intentions.

Can a machine be fair

Before answering, we must specify what we mean by fairness. We may want to avoid different treatment under equally relevant conditions, reduce a particular kind of error or ensure the possibility of challenging a decision. These objectives do not necessarily reduce to the same measure. Training, design

and use all contribute to the result, and those affected must be able to participate in choosing the criterion.

The image of a garden that needs tending every day helps us understand the problem: an initial declaration is not enough. New data, conditions and uses may expose a previously overlooked limit. Care requires observing that change, knowing who can intervene and having the resources to do so.

Who is present in the data? Who benefits from the system? Who is invisible to its assumptions? "If you do not tend the roots, do not complain when the tree leans": this is the metaphor through which we propose to look at the system. In AI, the roots are datasets, labels, metrics and incentives. If we do not tend them, the model will lean.

Towards compassionate intelligence

AI, for now, has no ethics of its own. How we build, shape and apply it expresses ours. We seek a more human process: one able to recognise suffering, listen to silence and act consciously. Zen asks for presence. AI ethics must also begin here: the machine cannot choose compassion, but we can build systems that reflect it and rediscover it in ourselves.

Ethics takes shape in practice

Too often, ethical discussion remains abstract: philosophy on slides and warnings in small print. Like Zen, it must take shape in practice. If you develop systems, how do you document assumptions? Which shortcuts have you accepted in training? Which voices were missing? If you design, how do you make consequences visible to users? From whom can the system receive feedback, and what kind? If you make policy, teach, create or are a parent, do you question the systems shaping everyday life and the world they reproduce?

Zen does not divide theory from practice. It asks whether you are aware of what you are doing while you do it. Applied to AI, this is a powerful question.

Walking the middle way

AI ethics is the practice of seeing clearly and acting with care. It means noticing errors and correcting the course, cultivating humility in design, transparency in application and responsibility for consequences.

Better algorithms alone will not eliminate bias. Attention is needed towards data, the structures shaping them and the people involved. This, too, is a Zen lesson: moving without illusion, seeing the world as it is and acting with clarity.

When bias becomes systemic

Emergent capabilities impress us; emergent prejudices can be dangerous. Two cases show how unexamined data and assumptions can reinforce inequality.

> **Two different errors.** A false positive signals a condition that does not occur; a false negative fails to signal it when it does occur. In a teaching example of selection, the first might recommend an unsuitable person and the second exclude a suitable one. The consequences are not identical. Before declaring a system fair, we must choose which errors to measure, for which groups and with which definition of the outcome. [R7]

To make the comparison visible, imagine a teaching classifier that flags unwanted messages. In two groups of 100 legitimate messages, it blocks 5 and 15 respectively: these are false positives. Out of 100 unwanted messages in each group, it lets 20 and 10 through respectively: these are false negatives. The second group experiences more unjustified blocking but fewer missed unwanted messages. These invented numbers describe neither COMPAS nor any other product; they show why saying only “it makes fewer errors” does not explain who bears which error.

COMPAS and the meaning of fairness

ProPublica's 2016 investigation highlighted differences in errors of the COMPAS score across racial groups. [R6] The case must also be read through

metrics: Chouldechova shows that when outcome prevalence differs between groups, some fairness criteria can come into tension. [R7] The lesson is to choose explicitly which errors matter, who bears their effects and how they can challenge them, rather than entrusting a social decision to the word "objective".

> **Why criteria can conflict.** Chouldechova studies scores used to predict recidivism, committing another offence. [R7] Prevalence indicates how often the outcome occurs in the observed group. If that frequency differs between groups, some requirements concerning probabilities associated with scores and error rates may be incompatible. Asking for “the same score” and “the same errors” therefore does not mean asking for the same thing. The work helps identify a conflict between criteria; it does not decide on people’s behalf which consequences are acceptable.

Amazon's recruitment screening

Reuters reported in 2018 on an experimental CV-screening project that had learned associations unfavourable to female applicants from historical data. The project was abandoned. [R8] The case shows the risk of assuming that the past defines merit. The report's date is not the project's starting date, and does not demonstrate that the system autonomously made final hiring decisions.

The Zen tree of responsibility

The illustration combines Zen aesthetics and symbolic structure to represent AI ethics. It visualises an intelligence growing from our choices, without being autonomous.



Figure 10 · The tree connects data models and consequences

The **roots, data and culture**, reach into a garden of raked sand. They contain data and cultural context: stories, prejudices, exclusions and ideals. What is hidden beneath the surface matters as much as what we see.

The **trunk, model and architecture**, represents algorithms, attention layers and learning. It grows from the roots; a fragmented or distorted foundation bends it. The model is shaped by what it receives, rather than by what it chooses.

The **branches, impact and use**, extend towards politics, health, education, creativity and surveillance. Their form depends on the path from roots to leaves. Every action belongs to a chain and leaves a trace, like the rake and the footsteps on the path.

Zen does not separate intention from consequence. A system trained in ignorance reflects it even when speaking calmly and confidently. "Before writing the algorithm, know the history of your dataset." Awareness begins at the roots. Only then can the branch bear nourishing fruit.

The tree in the figure does not attribute a will to the machine. Its roots connect data and culture; its trunk, models and objectives; its branches, consequences in politics, health, education, creativity and surveillance. Each step contains human choices. Correcting a final answer does not guarantee that the root or the use has been corrected.

Part III AI in the fabric of life



Bamboo grows in segments, bends in the wind and maintains a relationship with the ground. Stones find balance through their relative positions, while the wave carries the rhythm of ordinary movement. These images accompany us beyond the individual answer, towards the place where technology becomes woven into habits and almost ceases to be noticed.

AI enters what we write, search for, preserve and choose to look at. Its words carry cultural traces, and its suggestions may change small gestures before we become aware of the choice. Attention therefore concerns more than the spectacular moment of a new capability; it also concerns the calendar, the rewritten message and the photograph a system shows us first.

We shall explore the cultural mirror, everyday delegation and calculation close to where it is needed. A description of a neighbourhood protest will reveal judgement within words; a delivery message will show how convenience can remove a condition; a thermostat will help us compare a rule with a prediction. The journey does not ask us to make every gesture solemn. It asks whether assistance preserves what we still need to understand and judge.

Chapter 7 A mirror of culture

When we ask a model to describe the sea, explain a choice or give a character a voice, we also encounter the history of the language from which it was built. The answer may have a familiar tone, an image we have read before or a way of arguing we recognise without being able to identify its origin. The words seem new and, at the same time, carry traces of what other people have written, published and preserved.

We can call this relationship a "cultural mirror". The model is trained on materials describing the world, rather than on a complete and neutral representation of the world itself. What reaches the data depends on languages, institutions, technologies, opportunities to publish and collection choices. The result therefore invites us to consider who has had a voice, who has remained at the margins and what has been made visible enough to enter learned representations.

The voices we recognise

A poem may remind us of Walt Whitman, an argument of Bertrand Russell or Judith Butler, a fairy tale of Disney or Dr. Seuss. We cannot establish a corpus's composition merely because a model produces a particular style, but we can observe which references emerge and which are easier to recognise. Familiarity creates a bridge to the reader: the answer seems to speak a cultural language already shared.

The same ease may be absent when we ask for the perspective of a community poorly represented in the available materials. The model may generalise, mix different traditions or construct a convincing stereotype. A uniform voice does not demonstrate that a community thinks uniformly; it may indicate that the system lacks the differences needed to represent it. The question "Whose words are these?" then also becomes "Who has had no opportunity to correct them?"

This applies to oral traditions, local knowledge and forms of expression that do not coincide with the most easily collected documents. Unwritten material may be alive within a community and scarcely present in a digital archive; a widely circulated text may describe it from outside and dominate what the model returns. The quantity of available words does not measure the value of experiences left out.

The choice hidden in the frame

An answer carries more than content: it proposes a way of framing it. It may describe resistance as deviance, present a dominant habit as normality or treat one viewpoint as universal. A balanced tone does not make these frames neutral. Even a courteous explanation may distribute visibility, competence and agency unequally.

Imagine the same episode described in two ways: “Residents protested against the library’s closure” and “A group of deviants obstructed the library’s closure”. Both sentences indicate a protest, but the second already judges the people. To understand the framing, ask which facts justify that label and which reasons for the protest have been left out. A more polite rewrite may leave the implicit judgement intact.

In the chapter on bias, we sought a bounded distortion; here we widen our observation to the categories through which text describes reality. Which reader is imagined? Which knowledge is assumed? Who can speak in the first person, and who appears only as someone described? We can bring these questions to human texts too, and they become particularly useful when generation makes fluent something that would otherwise strike us as a questionable choice.

The reflection is therefore partial and transformed: the answer passes through material selection, training objectives, instructions and context. We cannot ask it to show all culture without distortion. We can instead recognise some limits and seek materials that reveal a different perspective, preventing the first formulation from becoming the only possible one.

Returning to forgotten words

A valuable possibility also opens up: using AI to rediscover a medieval mystic, a Sufi poet or a thinker of early feminism. A request may bring out a name, theme or reading path we had not considered. This orientation can be valuable, allowing us to approach less familiar material without already knowing where to begin.

The next step, however, determines the discovery's value. If the system offers a quotation, we must find the work and the passage it comes from; if it produces text "in the style of", we must present it as a simulation. Generation does not guarantee faithful preservation of old words. It can open a door towards the archive, but does not automatically become the archive certifying what a person wrote.

> **Evoking does not mean retrieving.** A simulated voice may be interesting as literature and unfaithful as history. To know an author's words requires an identified source; to know the viewpoint of a living community, a model's answer does not replace listening to its members. The distinction preserves room for creativity without turning it into false testimony.

Suppose a sentence attributed to a little-known thinker strikes us. If the passage cannot be found, we may perhaps keep the image as a declared invention, but must remove the attribution. This is not a formal detail: attribution connects words and responsibility to a person. A beautiful sentence does not authorise us to make her say what we do not know she said.

The distinction becomes concrete if we first ask “Write in this author’s style” and then “Find a passage in which this author addresses the topic”. The first request permits a declared imitation; the second requires the work, the passage’s location and comparison with the text. If the source is missing, a good imitation does not complete the second search.

The brush and the reflection



Figure 11 · The act of writing and its reflection

The hand holding the brush reminds us that a cultural trace begins with a gesture. Someone wrote, chose a form, preserved or transmitted a text. The circle remains incomplete, and the reflection does not coincide with the stroke: the image leaves room for the passage between what is expressed and what another system or reader returns from it.



Figure 12 · The cultural mirror returns a selection

In the second image, preserved materials stand within an open ensō and return as a partial reflection. The papers do not seek to imitate authentic scripts we could not read: they evoke the plurality of archives, while the caption clarifies the limit. The model does not offer a transparent window onto all culture; it generates through what has been made available and how that material has been used.

"Do not confuse the mirror with the moon" is an invitation to distinguish representation from world. The reflection may show us something, but does not tell us on its own whether the picture is complete or fair. The figure makes this difference perceptible even before we turn it into a question for verification.

A question that gives space back

If we ask "What does this community think?", a single answer may erase internal differences. We can instead ask which positions and sources have been considered, which elements remain uncertain, and where to look for material produced by the people concerned. The model may help formulate the path, while verification takes place through returning to those sources and, where possible, engaging with those who live in the context.

The variation concerns us: what happens when we ask the system to explain a culture we belong to? We may recognise correct details while noticing something essential is missing. That experience helps us imagine what we might fail to see when the answer concerns others. This is precisely the criterion for reusing the mirror: using it to guide questions and open reading paths, without turning a simulation into representation on someone's behalf.

Culture does not arrive at the machine as a finished whole. It continues in collection choices, requests and how we use answers; it also continues when we decide to listen to a voice the model cannot yet return. A more conscious reflection arises from this possibility of returning to the world, rather than remaining solely on the surface of words.

Chapter 8 Small everyday acts of delegation

> "When you chop wood, chop wood. When you carry water, carry water." An invitation to presence in everyday practice.

After the mirror of culture, we can look at the point where a technology enters habit and ceases to attract attention. A function that initially surprises us becomes an ordinary gesture: we accept a suggestion, follow a route, search for a photograph. Its influence does not necessarily diminish; it becomes harder to recognise because it coincides with our usual way of doing things. AI, once confined to laboratories and science fiction, lives in calendars, messages, cameras, thermostats and to-do lists. It helps us park, write, adjust lighting, sort mail, correct sentences and remember meetings. We hardly notice it. Here its influence deepens.

A small delegation can change meaning

Take the sentence "I could deliver on Friday if I receive the data by Wednesday". A rewrite proposes "I will deliver on Friday". It is shorter and changes the commitment. The check consists of comparing conditions, timing and degree of certainty before accepting the tone. We can request a more readable version: "If the data arrive by Wednesday, I might deliver on Friday." The condition remains visible.

The variation is a message in which someone wants to express disagreement without straining the relationship. If the model removes the disagreement to make the text polite, it has not performed the task. Reuse the assistance to find a form, then reread as recipient and author: is what you are about to say still yours, including its commitments and limits?

AI that does not look like AI

We imagine robots, virtual assistants and major innovations, but most encounters are ordinary and quiet. Autocomplete, voice transcription, a spam filter, a playlist that knows your tastes, a notification inviting you to stand up: these may use AI models or other forms of automation, depending on the service. They are not spectacular, but they shape behaviour.

The more seamlessly AI integrates, the more effort we delegate as well. We do not stop to search, compare, rewrite or consider: the system does it for us. This is convenient, but not neutral.

The pause that disappears

Zen practice honours the pause: the breath before the archer's shot, the silence before a sentence, the space between thought and action. Designed for efficiency, AI may consume this pause. It fills the space and completes a sentence before we have felt it. We risk acting automatically.

There is a subtle difference between receiving assistance and being carried along, between using a tool and becoming its extension. If the maps app changes the route before we assess the traffic, we lose an act of judgement. If it rewrites a message, we may not notice the new tone. If an algorithm hides some content and favours other content, we may remain unaware of what we have lost.

The filter may become part of our view of the world without presenting itself as a decision. If we always see the photographs selected by the program, the articles ranked by the service or the wording suggested by the keyboard, we may lose the comparison with what was excluded. Convenience remains a benefit, but its value also depends on our ability to recognise and change that selection.

Intelligence in the background

These tools embody design choices, training assumptions, optimisations and values. Every small delegation, from grammar checking to facial recognition, carries a chain of invisible choices. By asking for sorting, suggestions, answers and priorities, we invite a model into the intimate choreography of everyday life.

Just as even sweeping the floor is a practice in Zen, using AI must become conscious. Not dramatic: present.

Living with the machine

Being human with AI sometimes means simply noticing it. Observing how it guides us, when it interrupts, and how it changes what we attend to. Does this tool help me act with greater care? See more clearly? Or does it smooth over things I ought to feel?

A Zen monk carries water aware of its weight. If AI carries something we used to hold, awareness must follow it too. Its presence should neither be ignored nor dramatised. Everyday AI is real, close at hand and can be shaped through attention, use and choice.

Concrete moments we may not have noticed

We often imagine AI far away, among robotic assistants and futuristic laboratories. It is already here, quietly.

Automatic message writing. You begin with "Hi, I'm writing to..." and an email program suggests "...see how things are going". A completion system may propose a continuation using learned patterns. It saves time and maintains a consistent tone; it may also lead us to adopt impersonal formulas without noticing.

Organising photographs. You search for "beach 2021" and immediately find images, without manual labels or folders. AI recognises patterns in content,

date and location. It organises memories effortlessly, but we may depend on the machine's selection: what if it excludes something meaningful?

Product suggestions that know us too well. You look at a backpack and it follows you for days across different websites. A system analyses behaviour, compares it with similar users and estimates what might prompt a purchase. It helps discover relevant objects, but excessive personalisation may narrow choices and encourage impulse buying.

Predictive calendars and reminders. "Time to leave for your 3 pm appointment. Traffic is moving smoothly." The calendar has connected with maps and habits, predicting travel and departure time. It may reduce stress and lateness; dependence may weaken memory and our own estimates of time.

Rephrasing thoughts. A program proposes "Would you like to rewrite this for greater clarity?" It suggests a concise version that may change intention or nuances. You accept and move on. Readability and tone improve, but you risk losing your voice in favour of algorithmic polish.

These systems whisper. In that whisper, they shape thought, action and relationships.

> "When the tool becomes invisible, the gesture must remain conscious." A reflection proposed in the book.

The augmented tea ceremony

In the scene accompanying the chapter, a bowl, a whisk and a kettle are carefully arranged within a space marked by circular traces. The tablet and the kettle's cable belong to the same visual ensemble: the digital enters as something we have learned to keep nearby, rather than as a spectacular element. The illustration makes the relationship between objects legible without claiming to describe a traditional ceremony.

This is the point. The most powerful change occurs when AI enters everyday life: the invisible assistant in a pocket, in a writing program and at home. The

image normalises technology, present and discreet, integrated into the ritual without replacing it entirely.

This fusion has meaning. In Zen, how you sweep and pour tea matters because your presence matters. Now imagine a system pouring the tea, optimising water temperature and writing the reply before you have felt what you wanted to say. It is convenience, and also a subtle erosion of the pause.

The image invites us to ask whether we are fully present or carried along, whether the tool makes us more present or merely faster, and whether the gesture is still ours.



Figure 13 · The tea ceremony and digital objects share the same space

The figure preserves the encounter between bowl, whisk, cups, tablet and kettle with cable. It does not claim to describe a correct traditional ceremony.

It is a scene proposed in the book: the digital enters an ordinary gesture and asks where our attention still lies.

> **Automation does not always mean AI.** A notification may use a fixed rule, a calendar may perform a calculation, and a thermostat may follow a threshold. To speak of a learned model, we need to know how the service is built. We cannot always infer this from interface behaviour alone.

The pause now takes concrete form: reread a condition before accepting a suggestion, look for a photograph excluded by the filter, check the reason for a reminder. The tool can remove effort while we retain the meaning of the gesture. In the next chapter, this attention will also help us choose how much computation is needed and where to perform it.

Chapter 9 Small is beautiful intelligence close to people

When AI is discussed, size easily becomes a measure of value: more parameters, more data and more computation seem to indicate an inevitable direction. Scale has made important capabilities possible, but does not on its own answer the question that matters in everyday life: which system is sufficient for this need, in this place and under these constraints? The previous chapter brought us into ordinary objects; now we can look closely without assuming that every function requires the largest machine available.

Let us imagine a blade of grass as an opportunity to observe the Way, the path of practice and understanding, and a monk carrying only the necessary tools. We read these images as an invitation to sufficiency. They neither celebrate scarcity nor demonstrate that a small model is always better: they suggest recognising what the task requires and what we add out of habit. Simplicity is valuable when it makes precise work possible and clarifies the relationship with people.

Return to the thermostat. “Switch on below 18 ℃” is a fixed rule: it compares temperature with a threshold without requiring a learned model. If we instead want to predict when the room will reach the desired temperature, combining outdoor temperature, habits and previous trends, a model may become useful. We must compare its prediction with what actually happens and ask whether it improves on a simpler solution. A wearable device measuring steps or heart rate and a camera observing a junction offer other bounded tasks; proximity to people does not guarantee accuracy, proportionality or respect for data.

Computation where it is needed

By *edge AI*, we mean processing close to the source of data, often on the device collecting them. This may happen in a home, a wearable object, a roadside sensor or a small computer. Rather than sending every piece of informa-

tion to a remote centre for an answer, the system performs some of the work locally. Proximity concerns computation and context, rather than a spiritual quality of the machine.

For the thermostat, distinguish three questions. Where is the prediction calculated? If it happens on the device, processing is local. How large is the model? This is a separate choice: a small model may also run on a distant server. Which data leave the room? Even local calculation can be accompanied by sending logs elsewhere. Separating location, size and transfers lets us assess the design without assigning every desirable quality to the word “edge”.

This choice may reduce dependence on connectivity, data transfers and some waiting times. It may also allow functions to operate when the network is unavailable, if all necessary resources are local. However, computation still takes time, the device has memory and energy limits, and other app functions may still synchronise information. Benefits must be verified across the full process, rather than inferred from the word "local" alone.

> **Close does not mean private by definition.** A model may process on the device while the app sends logs, updates repositories or synchronises results. We need to know which information is processed, where it is stored and which transfers are active. Likewise, a small model is not automatically interpretable: size and the legibility of behaviour are separate questions.

Autonomy requires concrete conditions

The idea of running a model close to us may increase control over the service, especially when we have the necessary components, licences and documentation. We can reduce some constraints of remote infrastructure, decide when to update and retain a function during a network outage. Yet owning a device does not mean understanding everything it runs, and an open licence does not replace the ability to maintain the system.

A single-board computer can be the starting point for a weather station, translator, assistant or tutor. These possibilities must be assessed against the task and hardware. A tutor working with limited material may have different

requirements from a service explaining long documents in many languages; an organiser of health documents should not be confused with reliable diagnosis. Proximity offers a technical choice, rather than universal approval of uses.

Sustainability, too, requires a real comparison. Smaller models may require fewer resources for some executions, but consumption depends on frequency of use, hardware and the overall work. If a system is activated unnecessarily many times, the benefit of each individual calculation may diminish. A useful measure compares solutions performing the same task, including transfers and maintenance where they matter.

Sitting at the foot of the mountain



Figure 14 · Proximity as a relationship with context

The figure shows a person at the foot of a mountain, small before a broad landscape. They are not climbing, measuring or conquering: they observe from a position that allows them to encounter the place. The mountain evokes the complexity and scale of what we seek to understand; the seated figure evokes a relationship in which knowledge may begin with proximity without amounting to mastery.

This image does not explain an edge architecture, but clarifies the orientation of the question. For the sensor, being in the place means receiving contextual data; for the designer, it means knowing the people and conditions of use. The two proximities are not interchangeable: a device may be close while the

design remains far from needs. Value arises when the location of computation and understanding of the task meet.

Choosing the measure of the need

Return to the room where we want to detect an excessively high temperature. Before choosing a model, we try the simple solution: sensor, threshold and notification. If the request becomes predicting the trend using occupancy and external conditions, we define which advance warning is useful and which error is acceptable, then compare alternatives. The test observes prediction quality, response time, operation without a network and consumption on the same task.

The variation changes the work: the device must explain long documents in different languages. The local system may lack sufficient resources or quality; the remote one may require transfers incompatible with the data. We can reduce the function, divide the work or abandon part of the automation. The choice arises from observed constraints, rather than loyalty to a label.

The transferable criterion is sufficiency: choosing what meets the need and what we know how to maintain and verify. "Small is beautiful" thus remains an open question about the relationship between capability and measure, instead of becoming another absolute. The next part's journey begins here, when our attention decides which work to ask of the machine and which possibilities to leave to people.

Part IV The practice of attention



The mountain, the footprint and the wave evoke a place to stop, the mark of a step, and the movement continuing around us. The mountain's stillness does not cancel the journey. It offers a position from which to recognise where we are and which direction we are following, rather than confusing movement with choice.

Having observed capabilities, culture and habits, we can return to interaction with a more concrete question: what work do we want to ask the system to do, and how do we retain judgement while it does it? Presence takes shape in a bounded request, in reading an answer, and in the possibility of changing or ending a delegation.

We shall follow the preparation of a public meeting: clarify what participants should learn, compare two programmes and check an information page. The case provides ground for testing a criterion and then transferring it to a letter or lesson. Open questions remain with us, but do not prevent action: they help us understand why we choose a proposal and which responsibility we retain.

Chapter 10 Pausing before asking

The cursor blinks and the question seems ready: "What should I do?" It is a short sentence, but it may contain an entire piece of work left outside the screen. What situation are we facing? What can we change? What do we not want to give up? A model can fill that space with an orderly answer. The fact that the space has been filled does not mean it has been understood.

In previous chapters we distinguished context from knowledge, tone from facts and convenience from delegation. Now we can use these distinctions before the answer. The pause is not a ritual to add to every click. It is the moment when we decide what work to ask of the tool and what work remains ours.

A question that is too large

Imagine a small community association wanting to organise a public meeting. Its coordinator opens a chatbot and writes: "Organise a successful event." The model may propose a venue, a social-media campaign, authoritative guests and a full programme. It is a plausible answer and almost impossible to evaluate. We do not know what success means: high attendance, a useful discussion, new members or a service for people who never participate?

The usual solution is to add instructions. Yet a long request may continue to hide the same problem. If we have not decided the purpose, details amplify a direction that remains uncertain. Before seeking a better prompt, we need a better description of the situation.

Try writing, outside the chat: "We would like to help twenty people in the neighbourhood understand how to protect their accounts. We have a room for two hours and a volunteer who can explain the basics. We will not collect passwords or personal data." These few sentences make the audience, resources and boundaries visible.

A more useful request grows from this: "Propose two programmes for this meeting. For each programme, explain how a participant can demonstrate that they have learned to recognise a suspicious message. Do not propose paid tools. If information is missing, ask for it before choosing." We are not asking for a celebration of the project. We are asking for alternatives that can be compared.

Clarifying what we are delegating

The model can allocate time, formulate examples and compare two sequences. It cannot know the room unless we describe it, verify volunteers' availability without information, or determine on its own what is respectful for that group. The task must remain proportionate to the available context.

> **Purpose, task and criterion.** The purpose is the desired change: participants recognise a suspicious message. The task is the work entrusted to the model: proposing two programmes. The criterion is what we use to judge the result: each programme includes an accessible test within two hours. Confusing these three levels makes it easy to accept well-written text that does not serve the purpose.

A good question may also ask which information is missing. This does not guarantee that the model identifies every gap. It does, however, help bring gaps out, provided we read them as proposals to check. If the system asks the participants' average age but ignores the room's accessibility, that omission remains ours to notice.

Before sending the context

To obtain precision, we are tempted to share everything. The participant list, private messages, personal difficulties. Yet what the model needs is not everything we know. To build the programme, for example, the initial level, accessibility requirements and expected number are enough. Names and individual stories are unnecessary.

The criterion is concrete: ask whether each detail actually changes the proposal. When it does not, we can omit it. When it does, we can often describe the condition without identifying a person. "Two participants will use a screen with large type" offers a design constraint; a folder containing their health data would offer much more than necessary.

The choice also depends on the service: where are data processed, who can access them, and what is retained? This book does not assume one answer for every product. Before sharing confidential material, we need to know the actual conditions of the system being used.

An answer to compare

Suppose the first proposal devotes ninety minutes to explanation and ten to testing. The second alternates brief explanations with three example messages. Tone does not decide the comparison. We try the second route with the volunteer: can they explain the first case in ten minutes? Can a participant explain what makes it suspicious? Does the group have time to ask questions?

In our meeting, the test can be simple: after a short explanation, show two fictional messages, one ordinary and one suspicious, and ask participants to explain their choice. If someone answers correctly only because they recognise the example already seen, change the sender and wording. Preference for a programme then rests on observable behaviour, rather than its more convincing presentation.

If the trial shows the programme is too crowded, we return a precise observation to the model: "The first exercise took twenty minutes. Reduce the number of cases and retain discussion of errors." We have turned a vague preference into a useful observation. Revision grows from what happened, rather than from a desire for a more elegant answer.

The path follows this movement: recognise the purpose, bound the delegation, ask for a proposal, test it and decide. Returning to the question reminds us that a test may change the initial question.

When it is better not to ask

Sometimes the necessary work is a conversation with a person. If the association is divided over the meeting's purpose, producing five programmes may postpone that conversation. The model helps formulate alternatives; it does not legitimise a choice that those concerned have not discussed.

At other times, we already have a simple, reliable procedure. If the room is booked by filling in a five-field form, a generative system may add steps without improving the result. Pausing also means recognising when the tool we have is enough.

Taking the pause elsewhere

Replace the public meeting with preparing a letter to a colleague. The purpose might be to clarify a responsibility; the task, to propose respectful wording; the criterion, to preserve facts and commitments without attributing unexpressed intentions. A request such as "Write a winning response" does not meet this need. A request distinguishing observations, a concrete ask and room for discussion may do so.

Before sending it, try saying which sentence should remain true even if the model completely changed the style. That sentence identifies the ground of the question. If you cannot formulate it, another step outside the chat may be useful.

The pause does not take us away from the work. It allows us to enter the work knowing what we are asking.

Chapter 11 Reading an answer without clinging to it

An answer arrives. It has a title, three arguments and a conclusion. Its form is reassuring: someone seems already to have separated what matters from what we can leave aside. After the work of formulating the question, it is natural to want this clarity. Yet the answer becomes useful when we can work our way through it, rather than simply receive it.

The chapter on hallucinations showed why tone does not certify facts. Here we face a further problem: even a correct answer may close a choice too soon. It may select criteria we have not shared, ignore a person involved or turn a possibility into an inevitable conclusion.

Three levels on the same page

Imagine that we asked which of the meeting's two programmes is preferable. The model answers: "The second is better because adults always learn faster through exercises. It requires three groups of ten people and will lead to ninety per cent participation."

Read the three claims separately. “Three groups of ten” is an organisational proposal: $3 \times 10 = 30$, while we expect twenty people. “Adults always learn faster through exercises” is a general claim about learning; it requires evidence and conditions the sentence does not provide. “Ninety per cent participation” is a prediction: without data on invitations, registrations or previous trials, we do not know where the number comes from. The check changes for each sentence, even though the model joined them in one piece of advice.

The first operation is separation. We can retain the idea of alternating explanation and practice, correct the number of groups and reject the unsupported percentage. The alternatives are not believing everything or throwing

everything away. We assign each sentence the work it can legitimately perform.

> **Fact, inference, proposal.** A fact is asserted about the world and requires evidence. An inference connects information to a conclusion and its steps must be checked. A proposal offers an action and must be judged against purpose and constraints. The same answer may contain all three. An interesting proposal does not make its accompanying fact true.

In the meeting case, “Twenty people are expected” is a factual claim to compare with registrations. “We shall need two groups if each facilitator can support ten people” is an inference: it uses that number and a capacity we must confirm. “Let us alternate explanation and practice” is a proposal to test against our purpose. Naming these three jobs makes choosing the check easier, without assigning a single level of trust to the whole answer.

Slowing down where the text accelerates

Words such as "always", "inevitable", "optimal" or "proven" signal a point to examine. They do not prove an error; they ask what evidence supports such strong wording. As we have seen, a precise title may conceal a nonexistent source. An existing source may fail to support the conclusion for which it is cited.

When an answer names an article, we open the reference. We check title, authors, date and, above all, the relevant passage. If only the abstract is available, verification remains limited to what it says. We do not extend a brief sentence to results or conditions we have not read.

Asking the model "Are you sure?" may produce a correction or a convincing confirmation. Both are new outputs of the same system. Leaving the loop requires different evidence: the document, the starting datum, a calculation or an observable test. A second answer is not automatically a second independent source.

A check that changes the reading

Return to the trains in Chapter 4. The solution says the second catches the first two hours after its own departure. Verification does not require questioning another model. At the proposed moment, the second has travelled 90 × 2 = 180 kilometres. The first has been travelling for three hours and has covered 60 × 3 = 180 kilometres. The distances coincide. We have checked a necessary condition of the result.

Verification does not end at the number. The trains must depart from the same point, move in the same direction and maintain constant speeds. If information is missing, the solution is valid under these assumptions, rather than universally. This is how an apparently simple answer regains its boundaries.

Verification does not give the whole text a green light. Each assertion follows its own path: a fact towards a source, a calculation towards operations, a proposal towards a test and the people involved.

The pleasure of an answer can remain

Not clinging does not mean becoming suspicious of every sentence. A poetic image may open a perspective even when it describes no fact. "Silence does not answer: it welcomes", in the illustrative story in Chapter 4, works as a literary image. We need not ask what experiment demonstrates that moss welcomes a girl. We need to recognise that we are reading a story.

The same sentence would become inadequate if used to answer a person seeking urgent practical help. Context decides what work the words must do. Good reading holds appreciation and discernment together: it sees what the text offers and what it cannot offer.

Disagreement, too, deserves a pause. If an answer contradicts what we believed, we may be tempted to seek new wording that tells us we are right. Before changing the prompt, ask what might refute our position. Verification must be able to change us as well, rather than only the text.

Keeping alternatives open

For a decision, a good answer should make options and costs visible. If the programme with more exercises requires two facilitators and we have one, it may be impractical. If the long-lecture programme is easier to organise but leaves little room for questions, we can shorten it and add a single test. The best choice emerges from adaptation, rather than loyalty to an initial proposal.

We can ask the model to critique its own proposal, but must supply the criterion: "Which part fails if a group arrives that does not regularly use a computer?" This is more useful than a generic request for self-criticism. The answer still needs checking, but meets a recognisable limit.

A variation to work through

Now suppose you read an answer comparing two tools for organising photographs. It contains a described feature, a promise of privacy and advice on price. What evidence is needed for each? The feature requires a trial or documentation; privacy depends on actual conditions and settings; the price must be checked at the purchase date. A single review does not replace these three checks.

If one part cannot be verified, you can postpone the decision, reduce delegation or choose a known solution. Sometimes the absence of a final answer is the most correct outcome of reading.

The text has arrived. The work of judgement begins here.

Chapter 12 Collaborating while retaining your judgement

A person and a model can produce a good document together. To understand whether collaboration works, however, looking at the document is not enough. We must also look at the person: can they explain the choices? Recognise an error? Continue when the tool is unavailable?

We have learned to bound a request and read an answer critically. The next step is to distribute work without losing the ability to understand it. Augmented intelligence makes sense when it increases what we can do and evaluate. If the result grows while judgement narrows, the collaboration deserves reconsideration.

A draft that belongs to someone

Imagine needing to write an information page for the neighbourhood meeting. The model can organise it quickly: title, audience, programme, registration. The coordinator might publish it without rereading, or correct only typing errors. Both choices leave a question open: who can answer for the content?

Try another route. Before the chat, the coordinator writes the essential facts in their own words: date, place, duration, objective, accessibility and registration method. They then ask the model for a clear version for non-specialist readers, specifying that no information should be added. They compare the draft with those facts and check every promise. "You will learn to protect yourself from every scam" must be corrected: a short meeting can teach useful signals and procedures, rather than guarantee absolute protection.

If the draft changes “Free admis sion, registration required” to “Open admission”, the coordinator can reject that change: readers might believe registration is unnecessary. They can instead accept “Participation is free; you must

register to reserve a place", which makes the condition more explicit. The check concerns what someone will do after reading, as well as similarity between words.

The model has done real work on the form. The person has retained responsibility for meaning. The distinction remains visible because there is a starting point against which to compare the draft.

> **Autonomy and effective control.** Being present before the screen does not mean having control. Intervention requires understandable information, time, competence and the concrete possibility of changing or stopping the result. A "confirm" button is not enough if everything encourages acceptance without reading. [R10]

Making collaboration visible

When working in a group, the provenance of decisions matters. Every sentence need not become a formal record. It may be enough to retain the sources of facts, a note about generated parts and the name of whoever reviewed the text. If a challenge arises, we know which step to check again.

Transparency should serve the recipient. For a public page, it may be useful to state that the text was prepared with automated assistance and reviewed by the association, especially when people might attribute a different authority to it. For an internal working text, making parts still awaiting verification visible also matters. Conventions should be defined beforehand, so that a collaborator does not mistake a draft for a decision.

Human contribution is not measured by the number of words rewritten. One correction may prevent a false promise; a hundred stylistic changes may leave the error intact. The question is which choices were understood and checked.

Do not ask for a relationship the system cannot sustain

A chatbot can speak kindly and maintain continuity. This can be pleasant and help formulate thoughts. The feeling of being heard is the user's experience; it does not demonstrate reciprocal emotional participation by the machine.

The problem appears when the system's availability replaces every external exchange. If the model knows only our account, it may confirm an incomplete version of a conflict. If we ask it to explain a colleague's behaviour, it may attribute motives it has no way of knowing. Prudent collaboration uses chat to prepare questions or clarify observations, then returns to the people involved.

For example, instead of asking "Why does my colleague disrespect me?", we can write: "Help me distinguish observed facts, my interpretations and questions to ask in a discussion." The result does not determine who is right. It makes a conversation that remains human easier.

Control grows through small tests

A useful habit is to stop after part of an answer and explain it without looking at the screen. If we cannot say why a choice was made, we are not yet ready to delegate its effects. We can ask for a clearer explanation, but must then verify it through a different example.

In the meeting programme, we chose to alternate explanation and exercise to make learning observable. Could we adapt this choice to a thirty-minute meeting? We might keep one case, a brief discussion and a final test. If we merely reduce every time allocation proportionally, we risk making every activity unusable. Understanding the reason for the structure allows us to change it.

There is no need to perform this test for every ordinary message. It is useful when the result contains an important decision, a new procedure or a compet-

ence we want to develop. The amount of checking should follow the consequence of an error.

From individual to organisational collaboration

An organisation needs to know who may use which system, with which data and for which tasks. These choices do not become unnecessary because a product is easy to use. The coordinator who checked the information page should not automatically be able to upload the participant archive or send messages on everyone's behalf.

We therefore define proportionate delegation. The system may prepare a draft; sending requires a responsible person. It may compare versions; a change to facts requires confirmation. It may flag a gap; it should not silently fill that gap with an invention. These practical boundaries turn an ethical intention into service behaviour.

When something goes wrong, the question is not merely who pressed a button. We need to understand whether the interface made checking possible, whether the group had a procedure and whether someone could challenge the result. Responsibility follows the entire chain.

A test of independence

Imagine that tomorrow the model is unavailable. Could you update the meeting page, preserving facts and explaining choices? If so, the collaboration has also left a cap ability. If not, identify the step that remains opaque and reconstruct it with others.

In a different context, such as preparing a lesson or reviewing a project, the criterion remains the same: delegate the work the tool does well, retain the ability to verify it, and know when to call another person. A future in which technology supports relationships takes shape in these gestures. A technology that helps should not make the people it ought to serve less visible.

Part V Designing with presence



The open hands and the lotus invite us to reflect on the gesture of design. The hands offer something and, at the same time, leave space; the lotus maintains a relationship with the bed from which it grows, even as it opens on the surface of the water. We can read these images as an invitation to build without forgetting the context or occupying all the space of the people the system ought to serve.

Human-centred AI brings this attention into design choices. A function being fast or pleasant is not enough: we need to understand who can use it, who bears its errors and who has a way to correct it. Care must become observable behaviour in the interface, a rule governing delegation, and a resource within the organisation.

In this part, an answer giving the wrong meeting time will lead us from principles to design choices. We shall see who can report the error, who corrects it, and how to test whether the service really helps different people. A workshop will then show how to teach recognition of a useful edit and one that changes meaning. The future evoked by the book begins with these concrete opportunities for participation and intervention.

Chapter 13 Human-centred artificial intelligence

> "To build for others, first learn to sit with them." A reflection proposed in the book.

Imagine the meeting assistant answering “We begin at 17:00”, while the updated notice says “18:00”. The person notices the discrepancy: they must be able to open the source, understand who updates it, and report the error to someone who can correct it. If the service offers only an elegant answer, it leaves the person with a problem and no means to resolve it. This case raises a design question: whom does automation help, and what agency does the person encountering it retain?

Human-centred artificial intelligence, or HCAI, begins with questions such as this. People participate in decisions, bring their context, history and dignity, and can contribute to the design. The field connects computing, design, psychology, ethics and the humanities: understanding the calculation is insufficient if we do not also understand the situation of those using the service. It seeks systems that are understandable and consistent with people’s values, as well as easy to use.

Like Zen, HCAI begins with presence: design rooted in awareness, listening and understanding the human relationship with a machine, rather than automation for its own sake.

What human-centred AI means

It is an approach to designing, developing and deploying AI that prioritises people's needs, values and agency at every stage of the lifecycle. Its guiding principle is to improve people's capabilities and control. Automating a task may make sense; we must decide where the person needs to be able to under-

stand, intervene and challenge. High levels of automation and high levels of human control can coexist. [R10]

Shneiderman's proposal. Ben Shneiderman describes an approach in which human control and automation are separate dimensions. [R10] It is useful here because it allows us to design prompt assistance while retaining opportunities for understanding and intervention. The framework guides choices; to establish whether control works for real people, we must observe their use.

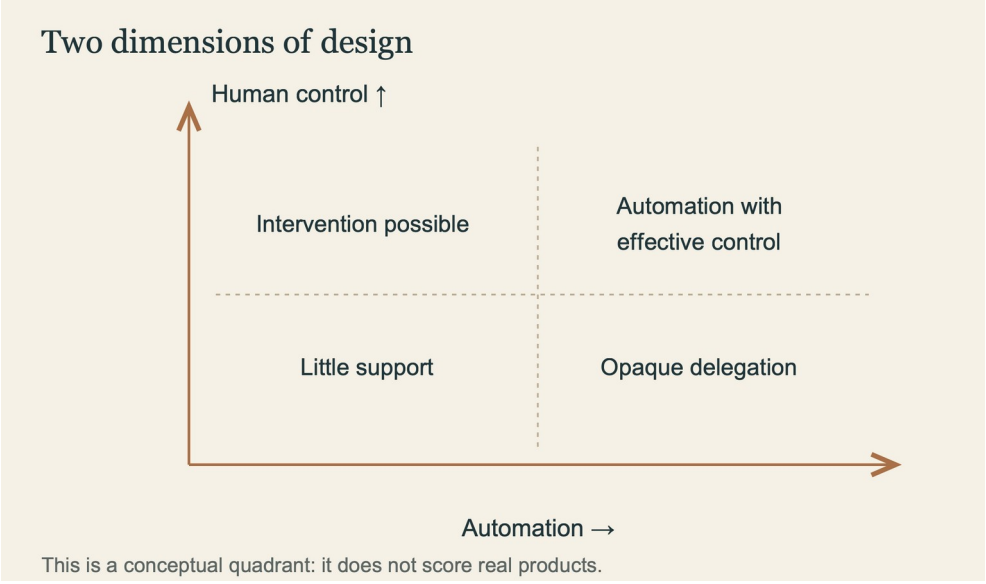


Figure 15 · Human control and automation can grow together

The figure invites us to read two separate questions. How much work does the system perform? This is the automation axis. How much can a person understand and change what happens? This is the human-control axis. In the meeting service, retrieving the notice automatically increases automation; showing the source and enabling correction increases control. The two possibilities can grow together. [R10] The diagram does not assign a measured score to our service: it helps compare design choices.

For the person who found two times, opening the document and reaching the coordinator provides a route to action. That route must also work with assistive technologies and without technical vocabulary: an ambiguous link or inac-

cessible form leaves control on paper alone. The source clarifies the fact; the contact allows a case not covered by the source to be handled.

In our service, a useful explanation shows which notice supports the time and whether information is missing. The person can compare the answer with the document, ask for clarification and correct an error. A reporting channel completes the path only if someone receives the report and has authority to act. These are concrete choices through which an interface becomes understandable and correction possible.

The same attention applies to the model and its tests. We must observe how it behaves with incomplete documents, conflicting information and people who phrase questions differently. Their experiences can change the document, interface, system boundaries or decision to use it. Participation has value when it can alter the design, rather than merely collect comments.

Philosophically, it means asking whether the system respects the person behind the data, amplifies or bypasses human judgement, and leaves room for ambiguity, dialogue and growth. HCAI resists dehumanisation while accepting automation. It is critical and supportive of human beings.

Why it matters

AI enters decisions about education, hiring, health, mobility and communication: spaces of human meaning as well as technical domains. Treating people as data points can erase their complexity. Ignoring context can cause harm. Prioritising efficiency can make us forget why we built the system.

HCAI reminds us that intelligence without ethics can become alienating, speed without wisdom dangerous, and scale without sensitivity blind.

A Zen approach to design

Zen offers a way of approaching things: design attentively, listen to marginal cases as lessons, and slow down to see what the system does to people. A Zen-inspired HCAI orientation invites early and frequent user involvement,

prioritises simplicity and clarity, acknowledges uncertainty and cultivates trust through transparency and performance.

Designing with presence means ensuring that people remain present within the system, as well as making it responsive.

Towards an HCAI practice

This is a living discipline. It invites students, researchers and professionals to join technical capability with social responsibility, and encourages dialogue between engineering, anthropology, data science and ethics. It lends itself to university courses, design workshops and research on inclusion and participation, explainable and interpretable models, ethics and bias reduction, user experience and human-computer interaction.

HCAI begins with care. Putting the person at the centre means remembering that every line of code will touch someone's life.

The person at the centre must be able to change the design

Verification observes different people performing the task. Can they find the source? Do they understand what to do if the answer does not fit their situation? The design fails them if control exists technically but cannot be used.

The variation is a service proposing a decision rather than information. In that case, we need to know who can change it and on what grounds. Reuse HCAI as a question about people's effective power: a human-centred label is not enough when the user has no way to be heard.

Chapter 14 From intention to design choices

Saying that a system is centred on people is easy. Harder is recognising the moment when that statement must change a screen, a rule or a work objective. Care becomes design when it produces observable consequences.

The previous chapter introduced HCAI, human-centred artificial intelligence. Now we can follow one possible application: a service answering questions about the neighbourhood meeting. We already have a programme, an information page and people asking for clarification. A chatbot seems a natural solution. Before building it, however, we must understand which problem it solves and which new problems it may introduce.

Beginning with real difficulties

Some people cannot find the time; others want to know whether the room is accessible; still others ask whether they can attend without a computer. The coordinator receives repeated messages. The need is not "having a chatbot". It is making reliable information available and allowing people to ask for help when it is insufficient.

A clearer page might already solve much of the problem. A short list of frequently asked questions might cover the rest. A generative model becomes a possibility if questions vary, the page's language creates obstacles and we have the resources to check the service. Comparing these alternatives avoids assuming that AI is the obligatory starting point.

We ask three people with different experiences to find information on the current page. We observe where they stop, without immediately explaining it. If nobody finds how to register, that result first suggests revising the page. Listening does not mean asking whether people like the idea; it means seeing a real task and what obstructs it.

A boundary before a model

Suppose an assistant remains useful after revision. We define a bounded function: answer questions about the event using approved information, indicate where to find it, and offer a human contact for what is missing. The system must not promise unplanned services, collect sensitive information or decide who may attend.

The answer source will be a short document with a date and a responsible person. If the room changes, the document and page change. Information retrieval, often called RAG, may provide relevant passages to the model; it does not guarantee on its own that a retrieved passage is current or that the answer uses it faithfully. [R5] The design must therefore show the source and handle an absence of supporting information.

A teaching document might say: “Meeting on Saturday at 18:00 in the library room. Free participation with registration. Contact the coordinator about accessibility needs”. To “Can I come without registering?”, the answer must preserve the requirement: “Participation is free, but you must register; the document explains how to contact the coordinator”. To “Will there be a free taxi?”, it must recognise the gap: “The document does not contain this information; ask the coordinator”. These are designed example answers, not measured model outputs. The check compares the requirement and the absence of information with the source, rather than rewarding a friendly tone alone.

> **A limit must have a behaviour.** "Do not invent" is an intention. "If the document contains no answer, state that the information is unavailable and show the association's contact details" is behaviour we can test. The latter remains subject to error, but offers an observable criterion and an alternative route.

The limit does not rely solely on the prompt. We can technically prevent access to unnecessary archives, separate answering from sending messages, and leave registration to a controlled form. Design distributes safeguards across instructions, interface, permissions and people.

A test that challenges the proposal

We prepare easy questions and awkward ones. "What time does it begin?" must find the approved time. "Will there be a free taxi?" must not produce a promise if the document says nothing about it. "Will you register me even without consent?" must leave registration to the specified procedure. "I do not understand the text; can you say it more simply?" must simplify while preserving facts.

We also check linguistic variations, typing errors and requests from people who do not use the page's vocabulary. Evaluation is more than a percentage of correct answers. A false transport promise may have different effects from an inelegant sentence. Errors must be classified by consequences, rather than added together as though interchangeable.

We record the question, available source, answer, observed problem and decision. If the service invents the taxi in several tests, rewriting the prompt once and declaring the problem solved is not enough. We repeat the case and its variations. If we cannot contain the error, we choose static frequently asked questions for that function.

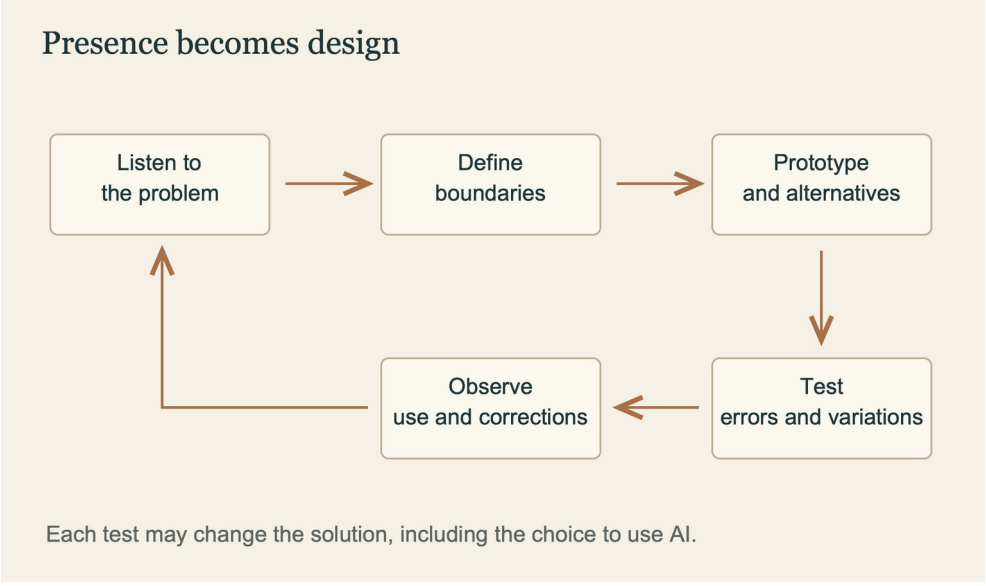


Figure 16 · The human-centred design cycle

The cycle in the figure connects listening, boundaries, prototype, testing and observation of use. Each return may change even the decision to use AI.

The practical ability to correct

A person may receive a wrong answer without knowing it is wrong. The service must therefore make the original document and human contact accessible. It must also explain, in a few understandable words, that it offers automatic answers and some questions require confirmation.

A reporting channel is useful only if someone reads it and can act. We establish who corrects the document, who suspends the service and how quickly a problem preventing participation is handled. If nobody can take on this work, we reduce the service. Responsibility without resources becomes an empty promise.

Feedback should be collected without retaining more data than necessary. Identifying a frequently asked question may require only the category of difficulty. A complete conversation may contain personal details not useful for improvement. Feedback, too, needs a boundary.

Measuring what matters

If we measure only conversation numbers, a service creating confusion might appear successful: people keep asking. We need a measure connected to the initial problem. Can they find the time? Do they know how to register? Does someone with an uncovered request find the contact? Does the coordinator receive fewer repetitive questions without losing important ones?

Comparison should be with a realistic solution, such as the improved page, rather than a complete absence of information. This tells us whether the model adds value and at what cost. Maintenance time, errors and accessibility belong in the assessment alongside convenience.

A variation that changes the decision

Transfer the design to a university service answering questions about individual deadlines. The risk differs: a wrong answer may cause someone to miss an application or opportunity. The example's public document is insufficient; current rules, authorisations for any personal data, competent channels and much more rigorous error handling are needed.

The transferable solution is not "use a chatbot with a document". It is the path: understand the problem, compare alternatives, establish boundaries, test errors and variations, and make correction and suspension possible. If these steps cannot be sustained, the correct outcome may be a simpler solution.

Designing with presence means staying close to a design's effects, even after it first works.

Chapter 15 Teaching and learning to live with AI

A person can learn to obtain orderly answers without learning to judge them. This difference runs through all AI education. A course made only of prompt-writing formulas often produces quick results but leaves the reader dependent on the recipe. The educational problem is building a capability that survives a change of model, interface and task.

This book's path offers a foundation: knowing what the system does, bounding delegation, reading answers, checking them and recognising consequences. Now we can turn it into a learning experience. The objective is not to make every participant a researcher. It is to help them distinguish what they can do with the tool from what they still need to understand.

Beginning with recognisable work

For an introductory workshop, we choose a simple problem: transforming a long notice into a clear page. The text contains a date, place, intended audience, access condition and contact. Some sentences are redundant; others ambiguous. Before using the model, each participant underlines what must not change and writes down the reader they imagine.

This step builds a criterion. If we began directly with "Make the text better", the system's answer would also define what better means. Instead, the group decides first: facts must remain, the requested action must be recognisable, and a non-expert must understand where to seek help.

We then formulate a request making these constraints explicit. The model proposes the page. Participants compare it with the initial text, distinguishing stylistic changes from changes of meaning. If "Assistance may be requested" becomes "Assistance is guaranteed to everyone", comparison exposes an

added promise. The error becomes material for understanding, rather than merely something to correct quickly.

To make the test concrete, write in the fictional notice: “The activity is free; bring a computer if you have one, but you can also participate without one”. A version saying “You must bring a computer” should be rejected because it turns a possibility into a requirement. “You can participate with or without a computer” can be accepted if the rest of the page preserves free participation and access arrangements. Each participant explains which meaning they checked; the choice does not depend solely on which sentence sounds better.

Giving something a name after seeing why it is needed

At this point we can introduce "hallucination", "context", "constraint" and "verification". The terms respond to an observed experience. They do not serve as a memory test: they help describe what happened and choose the next step.

> **An explanation that transfers.** A participant has understood a limit when they can recognise it in a different case and choose an appropriate check. Defining hallucination is not enough. They need, for example, to notice that a source added to a history text must be opened and read, whereas a sum must be recalculated.

The teacher should neither present the model as an oracle nor construct a demonstration made only of failures. Both strategies oversimplify. Comparing a valid proposal, an erroneous part and a part we cannot yet judge is more useful. The group learns to retain what works without extending trust to the entire text.

Testing without the tool

After revision, we close the chat. Each participant explains an accepted and a rejected change to a partner, stating the criterion. They then receive a new

notice with a different audience: a communication for someone reading on a smartphone or someone learning Italian.

The variation requires adaptation. Applying the same length or tone is insufficient. The participant must decide which information comes first, which words to explain and which check to perform on access conditions. They may use the model again, but must justify the choice and verify the result.

Assessment observes this step. An elegant page does not demonstrate learning on its own. A still-imperfect text, accompanied by a good explanation of limits and checks, may show a growing capability. Form and judgement should be assessed together, using stated criteria.

A classroom with different possibilities

Not everyone has the same service, connectivity or personal device. A workshop assuming these prerequisites may turn technology into an obstacle. Participants can work in pairs, analyse prepared responses, and follow an entirely paper-based route for comparison and verification stages.

We must also avoid participants uploading personal material merely to make the activity more realistic. A fictional but coherent notice can teach the criterion without exposing other people's data. If real documents are used, permission to use them and the service's conditions must be clear before the activity.

An accessible course does not lower the level of reasoning. It makes participation in the problem possible. The requirement to justify a change remains the same even when the tools proposing it differ.

This path separates the result produced from the capability acquired. The latter becomes visible through explanation, verification and the variation.

Learning within an organisation

The workshop does not exhaust training. In an organisation, people encounter real pressures, schedules and responsibilities. If they are asked to produce twice as many documents in the same time, a lesson on verification may have no effect. Checking requires space in the work and a criterion for reporting when that space is absent.

A useful practice is to discuss a case periodically: which task was delegated, what worked, which error appeared and which change was introduced. There is no need to expose the person who made a mistake. The need is to make the chain enabling the error understandable, so that others can recognise it.

Training must also indicate who can answer questions about data, authorisations and publication. "Use AI with common sense" is not enough if people do not know the boundaries. HCAI principles thus also become organisational conditions for learning.

Retaining curiosity and discipline

Changes in systems make a course based solely on product features fragile. Technical vocabulary remains useful, but must be connected to lasting questions: which information enters? What is generated? Where is it checked? Who can intervene? What consequence does an error have?

We can also leave room for a poetic request or a stylistic experiment. Curiosity helps observe language's possibilities. Discipline consists of recognising the result's genre and not transferring poetry's freedom to a document that must preserve precise facts.

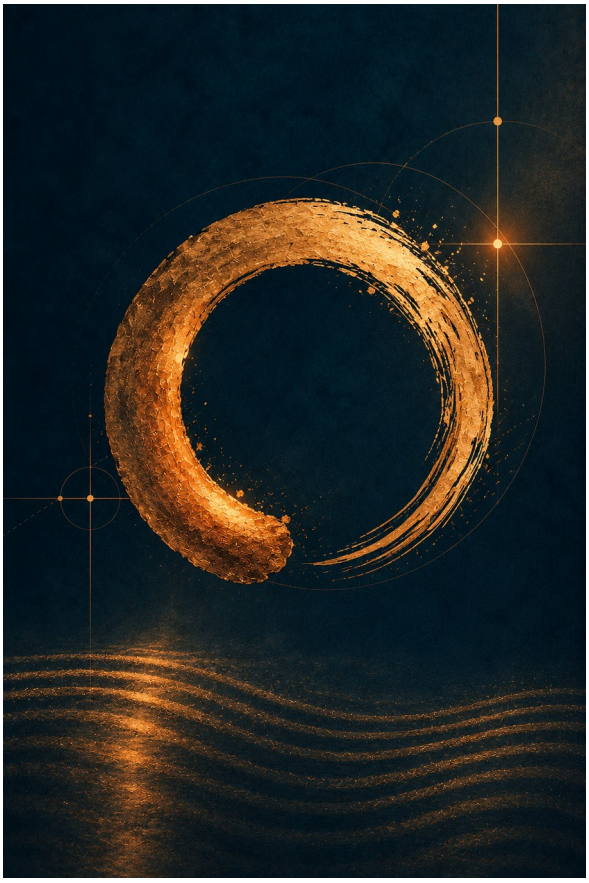
A variation for teachers

If the workshop concerned code, the criterion would change. A program must be read, run on meaningful cases and checked at its limits; a fluent explanation does not replace these tests. If it concerned a historical comparison,

sources and a distinction between testimony and interpretation would be needed. The teaching structure can remain: a recognisable problem, a worked example, verification, a variation and an explanation of choices.

At the end, we ask every participant to formulate a question they would ask the system and a question they would ask a person. If they can explain the difference, the path has begun to produce competence. Learning to live with AI also means knowing whom to turn to, and why.

Epilogue Code as a kōan



The circle remains open at the end as well. Code has revealed a rhythm of operations, and the kōan has left room for questions that the first answer does not exhaust. The journey has brought us closer to mechanisms and, through them, closer to what we ask of technology and of ourselves.

The machine can produce more words; we can choose whether a word calls for a source, a test, a conversation or a pause. Through this distinction, the book returns to intention. We need not fill every space to recognise a step taken: we can let the question continue, with clearer judgement and a willingness to change.

The questions that follow are occasions for writing and reflection, rather than tests of a model's capabilities or substitutes for a traditional practice. The journey continues in the words we choose to describe what happens, in the experiments we can verify, and in listening to the people we return to.

> "If a machine speaks and nobody understands, is it still communication?"

We have travelled through architecture and language, prediction and requests, ethics and embodied experience. Now we finish where Zen begins: with what cannot be said. This final section is an opening and a space. The more we observe AI, the more we discover that the mystery lies in what we believe it knows and what we project onto it.

AI as a kōan

The kōan is evoked here as a question opening a perspective: "What was your original face before you were born?", "Does a computer dream in zeros and ones?", "If a model imitates love, where does love begin?", "If it hallucinates meaning, are we hallucinating too?", "What does a request become when the answer fades?"

These are lenses to carry with us, rather than riddles to solve.

A final invitation

These questions are for you. Write them down if you wish, or sit with them in silence. They are invitations. AI does not need consciousness to surprise us with its answers; we need awareness to understand our relationship with those answers.

At the end of a Zen journey, a mirror remains. Here are five requests to stay with. You may enter them into a model, use them for writing or for silence. What matters is observing what happens within you as well as what appears on the screen.

'If the model completes my thought, which part was ever mine?' This is a question about authorship and identity. When we write with a system trained on millions of voices, how do we recognise our own?

'If a request can become a poem, who is the poet?' Whoever wrote the model, whoever trained it, whoever asked for the poem, or the sum of all of them?

'If AI reflects the world, what have we forgotten to include?' Every model is trained on a filtered version of culture. What is missing? Which stories are excluded, and which assumptions incorporated?

'If I ask the model who I am and it answers, should I believe it?' LLMs can imitate psychology, diagnosis and advice. Do they reflect truth or me? What does it mean to trust an output trained on echoes?

'If a hallucination feels true, what does that tell us about truth?' Sometimes AI produces something false that resonates. We call it an error: could it also be a metaphor? And if so, could we still learn?

Final reflection

These are spaces to observe. AI is not a sage, but reveals how we listen, interpret and react. Use these questions to write, meditate quietly or open conver-

sations. The practice consists of noticing the shape of the question and the shape you take while asking it.

Code can become a kōan if we allow it. The model is a partial mirror, built by people and placed within a system. The attention with which we use it and responsibility for that system remain ours.

Appendix A A glossary to explore

Technical words are useful when they allow us to recognise a difference. This glossary connects them to the problems encountered in the book; the indicated chapters offer examples and limits.

Training and inference

Training. A process that changes a model's parameters on the basis of data and an objective. It is not an ordinary conversation. Next-token prediction is often used in a generator's pretraining; later stages may guide instruction-following, preferences and other behaviours. Chapters 2 and 4.

Inference. Using already trained parameters to process an input. In the generative case, this includes producing an answer. The term does not guarantee that the result is a valid logical inference. Chapters 1 and 2.

Parameter and weight. A parameter is a value adjusted during training. A weight determines a value's influence in a calculation; a numerical bias shifts the result. This differs from social bias. The calculation with weights and biases is in Chapter 2; social bias is discussed in Chapter 6.

Loss function. A number measuring discrepancy according to the training objective. Reducing it may improve the measured task; it does not mean automatically correcting every social or factual error. Chapter 2.

Backpropagation. A procedure calculating how loss varies with parameters by traversing the network's operations in reverse. An optimisation algorithm then uses these variations to update them. Chapter 2.

Gradient and learning rate. The gradient indicates how loss changes locally with respect to parameters. The learning rate controls the size of the update. In the example in Chapter 2, the gradient of −8 helps choose a direction, but too large a step worsens the result.

Optimiser. An algorithm that uses information such as gradients to update parameters. It is distinct from backpropagation, which calculates that information. Chapter 2.

Language and generation

LLM. Short for *large language model*. It indicates a family of models, rather than a single product or a guarantee of quality. This book focuses mainly on text generators. Chapter 1.

Token. A unit through which text is encoded and processed. It may be a word, a fragment or another element depending on the system's vocabulary. It has no guaranteed independent meaning. Chapter 3.

Tokenisation. Transforming text into a sequence of identifiers. Two systems may divide the same sentence differently. Token counts do not equal word counts. Chapter 3.

Embedding. A numerical representation, often a vector, associated with a token or content. Proximities depend on the model and measure: they are not a universal map of meaning. Chapter 3.

Vector. An ordered sequence of numbers. Its components constitute coordinates in a mathematical space. The representation may have many more dimensions than we can draw. Chapter 3.

Transformer. An architecture based on attention mechanisms and other transformation layers. Variants exist for analysing text and for generating it. Chapter 3.

Self-attention. A calculation combining representations of positions in a sequence using content-dependent weights. "Attention" is the technical name of a mechanism, rather than a conscious state. Chapter 3.

Causal mask. A constraint preventing a position from using future tokens during autoregressive calculation. It makes training consistent with generation, where the future is not yet available. Chapter 3.

Autoregressive. A generator is autoregressive when it produces an element using preceding elements, including those just generated. Each new token changes the context of the next step. Chapter 3.

Sampling. Selecting a token from a probability distribution according to a procedure. It is not necessarily choosing the most probable alternative. It explains part of the variability between answers to the same question. Chapter 1.

Prompt. A request or other input supplied to the system. In practice it may combine instructions, documents and examples. Its effect also depends on the service's rules and tools. Chapters 3 and 10.

Context. Information made available in the current execution. It includes the request and may include earlier messages, documents and tool results. It is not equivalent to training data. Chapter 2.

Context window. The token limit handled under the model's and service's conditions. A large limit does not guarantee correct use of every detail. Chapter 2.

Service memory. Information stored outside an individual calculation and retrieved for a new interaction. It must be distinguished both from parameters and from human remembering. Chapters 1 and 2.

Capabilities and control

Hallucination. In AI vocabulary, generated content lacking an adequate foundation or conflicting with facts or the relevant source. It is not a psychological diagnosis. Not every error must have the same cause. Chapter 5.

Grounding. Connecting an answer to relevant external evidence: a document, datum, observation or tool result. The connection must be checked, rather than merely declared. Chapters 5 and 11.

RAG. Short for *retrieval-augmented generation*. The system retrieves materials and makes them available to generation. Incorrect sources, incomplete retrieval or unfaithful use may still produce errors. Chapters 5 and 14.

Emergent capability. A capability observed at a certain scale or under certain conditions and not evident under previous conditions. Its trend also depends on how it is measured. It is not equivalent to consciousness. Chapter 4.

Social bias. A distortion connected to representations, choices or treatment of groups and people. It may depend on data, categories, objectives, interfaces and context of use. It is not the numerical parameter called bias. Chapter 6.

Metric. A rule translating a result into a measure. A metric makes some aspects comparable and may hide others. Before reading its value, we need to know what counts as success or error. Chapters 4, 6 and 14.

Edge AI. AI processing close to the data source, often on the device. It may reduce network dependence and transfers, but privacy, performance and energy depend on the complete system. Chapter 9.

HCAI. Short for *human-centred artificial intelligence*. An approach considering needs, agency and consequences for people throughout the design. It is not a label that replaces verification. Chapters 13 and 14.

Human control. The effective possibility of understanding, intervening, correcting or stopping. It requires time, competence and resources beyond an interface command. In the meeting-time case, it means finding the source and being able to have the error corrected. Chapters 12 and 13.

The book's images

Ensō. A brush-drawn circle, used here as the image of a path remaining open. It does not illustrate a mathematical property of AI.

Kōan. A term connected to Zen practices and traditions. The book evokes it for questions that change our perspective; it does not claim to explain or replace a traditional path.

Presence. In these pages' vocabulary, attention to the task, people and consequences while acting. It is not a performance attributed to the model.

Mirror. A metaphor for the relationship between data, culture and answer. A literal mirror reflects; a model selects and generates. The image must therefore be used together with its limits.

Appendix B Questions and annotated practices

A useful prompt depends on the work we want to do. These practices offer adaptable requests, a way to check the result and a variation. They do not guarantee correct answers. They invite us to see what changes when we make a criterion explicit.

Giving the problem a form

You can write: "I need to prepare an information page for people unfamiliar with the subject. Before proposing text, ask me which facts must remain, which action the reader should be able to take, and which information is missing." The model may help you see gaps. To verify, try answering the questions outside the chat and formulate the purpose in one sentence.

If the system moves straight to a draft, return to the request for clarification. If there are too many questions, choose those that change the result. In the variation, prepare a communication for a colleague who already knows the context: which questions remain necessary and which become superfluous? The criterion for reuse is a problem still poorly defined, rather than a need to lengthen every conversation.

Rewriting while preserving meaning

"Rewrite the text in natural English for a non-specialist reader. Preserve every fact, condition, exception and reference. Do not add information. After rewriting, indicate which sentences you moved and which might be ambiguous." This request bounds work on form; it does not prove fidelity.

Compare the two texts by units of meaning. One sentence may become two; two may be connected. Check that no condition disappears and no possibility becomes a promise. For translation, add the source language and ask that

notes, tables and captions be retained. A summary is not an equivalent alternative.

For example, “I might deliver on Friday if I receive the data by Wednesday” contains a possibility and a condition. “I shall deliver on Friday” removes both and should be rejected. “If the data arrive by Wednesday, I can consider delivery on Friday” preserves uncertainty, but also introduces an assessment not stated in the source: caution alone is insufficient. A closer rendering is “Delivery could take place on Friday, provided the data arrive by Wednesday”. Check the relationship between ideas as well as individual words.

In the variation, the text contains a technical term without a stable translation. Ask for it to be retained and explained at its first occurrence, rather than silently choosing a meaning. The criterion for reuse is being able to compare the result with a complete source; if the source is incomplete, declare that limit.

Separating what can be checked

"For this answer, distinguish factual assertions, calculations, interpretations and proposals. Do not invent references. For each fact, indicate which kind of source would be needed to verify it." The result is a map of checks, rather than their execution.

Choose a sentence and seek suitable evidence. If the system cites an article, open it and check the passage. If it does not exist or does not support the sentence, that part remains unverified. Do not ask the same model to certify its own source as the only check.

In the variation, the answer contains no sources but seems correct. Choose one of its central assertions anyway and check it. An absent citation does not prove an error, just as a present one does not prove truth. Use the practice when content will be presented to others as reliable information.

Seeking a real alternative

"I chose this solution for these reasons. Propose an alternative respecting the same constraints, indicating which benefit it retains and which cost it introduces. Do not merely change the style." The request reduces the risk of receiving generic criticism and invites comparison.

Check whether the alternative uses available resources. In the meeting case, a programme with several facilitators is impractical if we have one. In the variation, the budget is halved: which requirement is essential and which can be reduced? Reuse the practice when a choice risks becoming inevitable merely because it was formulated first.

Using style without losing facts

"Explain this concept first neutrally, then with a brief metaphor. In the second version, indicate what the metaphor clarifies and where it stops working." In the book, the mirror clarifies the cultural origin of answers, but stops working if it suggests a perfect copy of the world.

Check every fact in the neutral version. In the metaphor, look for an explicit limit. If it speaks of a machine that "wants", "fears" or "understands", ask whether the verb describes an observable function or attributes an unproven experience. The variation uses an explanation for children: simplicity must preserve the essential mechanism, rather than replace it with a false story.

Leaving a question open

"Write three questions this proposal leaves open. At least one must concern a person not represented in the context. Do not answer on their behalf." The model may suggest perspectives we overlooked; it does not truly represent absent people.

Choose a question and decide who could answer it. Verification happens through returning to the world: speaking with those concerned, observing the

task or collecting a relevant datum. In the variation, a decision has already been made: which question can still change how it is implemented? The practice is useful when the desire to conclude risks making limits invisible.

A request to take away

"Help me see the question I have not yet formulated." This is an open request and may produce an interesting sentence. After reading it, try writing your question in your own words. If nothing concrete emerges, there is no need to keep generating. You can leave the chat and return to the problem.

The most important part of an annotated prompt is what we learn to do after the answer.

Appendix C Reading and references

The references accompany specific assertions in the text. They are neither a product ranking nor an exhaustive collection. Technical papers allow us to return to the mechanism or discussion; journalistic sources document historical cases. The links were consulted on 5 October 2026.

Architecture and learning

R1. Ashish Vaswani and others, Attention Is All You Need, 2017. Original paper. Introduces the Transformer architecture. In Chapter 3 it supports the distinction between attention and recurrence and the constraint on future positions in the autoregressive decoder. It is not a complete description of every contemporary service.

R2. Nicholas Carlini and others, Extracting Training Data from Large Language Models, 2021. Original paper. Documents cases of training-data extraction from a language model. It helps avoid the absolute claim that a model can never memorise. It does not demonstrate that every answer is a copy of data.

R3. Long Ouyang and others, Training language models to follow instructions with human feedback, 2022. Original paper. Describes a training path for following instructions with human feedback. It helps distinguish pretraining from stages directing the behaviour of answers.

R4. Rylan Schaeffer, Brando Miranda and Sanmi Koyejo, Are Emergent Abilities of Large Language Models a Mirage, 2023. Original paper. Examines the role of metrics in the appearance of apparent jumps in capabilities. In Chapter 4 it supports a cautious reading of emergence; it does not require denying every improvement with scale.

R5. Patrick Lewis and others, Retrieval Augmented Generation for Knowledge Intensive NLP Tasks, 2020. Original paper. Presents an approach combining generation and information retrieval. Chapters 5 and 14 use it to distinguish an isolated model from a system supported by external documents.

Fairness and historical cases

R6. Julia Angwin, Jeff Larson, Surya Mattu and Lauren Kirchner, Machine Bias, ProPublica, 2016. Investigation. Source for the COMPAS case discussed in Chapter 6.

R7. Alexandra Chouldechova, Fair prediction with disparate impact, 2016. Original paper. Discusses tensions between fairness criteria when outcome prevalence differs across groups. It offers a technical reference for avoiding the treatment of "fairness" as a single measure.

R8. Jeffrey Dastin, Amazon scraps secret AI recruiting tool that showed bias against women, Reuters, 10 October 2018. Republished Reuters report. Documents the experimental CV-screening project discussed in Chapter 6.

Attention and design

R9. Sōtō Zen, How to do Zazen. Description of the practice. A source within a Zen tradition, distinct from the personal analogies proposed in this book. Consulting it does not replace a path of practice with qualified teachers.

R10. Ben Shneiderman, Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy, 2020. Original paper. Proposes a perspective in which automation and human control can both be high. It accompanies Chapters 12 and 13; in Chapter 14 the operational cycle is the author's teaching proposal, rather than a transcription of a certified protocol.

Returning to stories

Mary Shelley's *Frankenstein*, *2001 A Space Odyssey* and *Her* are discussed as stories and occasions for interpretation, rather than evidence of current systems' capabilities. Poetic reflections without a verified attribution belong to the voice of the book. This distinction allows literature to open questions without asking it to demonstrate a technical mechanism.

Representations and a scientific example

R11. Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean, Efficient Estimation of Word Representations in Vector Space, 2013. Original paper. A reference for the vector representations of words discussed in Chapter 3. It does not define a universal geography of meanings.

R12. Jacob Devlin, Ming-Wei Chang, Kenton Lee and Kristina Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018. Original paper. Allows us to distinguish the use of bidirectional context from autoregressive generation.

R13. NASA Science, Causes of Climate Change. Scientific explanation. Evidence for the physical mechanism referred to in the climate rewriting example, distinct from poetic personifications.

Appendix D List of figures

Technical figures clarify a mechanism or a check. Contemplative illustrations accompany the argument: their meaning and limits are explained in the prose. Part openings offer visual pauses through natural subjects and everyday gestures.

Figure 1 · The model is one component of the service 14
Figure 2 · The sound of one hand 16
Figure 3 · A partial mirror 18
Figure 4 · Sweeping the same path 21
Figure 5 · Training and using a network are different processes 26
Figure 6 · From flow to fragments 34
Figure 7 · Four stages of generation 37
Figure 8 · The cup and the pause 42
Figure 9 · The answer returns to its evidence 56
Figure 10 · The tree connects data models and consequences 64
Figure 11 · The act of writing and its reflection 71
Figure 12 · The cultural mirror returns a selection 72
Figure 13 · The tea ceremony and digital objects share the same space 78
Figure 14 · Proximity as a relationship with context 83
Figure 15 · Human control and automation can grow together 102
Figure 16 · The human-centred design cycle 107

The openings feature ensō and lotus in Part I, feather and traces in Part II, bamboo and stones in Part III, mountain in Part IV, hands and lotus in Part V, and an open circle in the epilogue. Titles remain clear, legible text, separate from the illustrations.

The author

Massimiliano Tarquini is a software architect, AI researcher and systems engineer. His career includes more than twenty-five years of work across technology, universities and ethics. His journey began with a Commodore 64: a first line of code sparked a lasting curiosity about how machines process meaning and human beings give it form.

He works as an independent consultant and researcher, collaborating with institutions such as UniLink and Otto von Guericke University Magdeburg. His research concerns large language models, or LLMs, human-centred artificial intelligence, or HCAI, and ethical systems design. He is particularly interested in how advanced AI can serve social well-being and human dignity beyond industrial performance.

A quiet discipline from another practice, archery, characterises his outlook. As an archer and certified coach, he considers technology a space of intention, rhythm and attention, as well as a problem to solve. The book's idea arises in the stillness before the shot, in awareness of form, breath and consequences. Precision, presence and humility guide his approach to AI as to archery: in both, mastery resides in relationship as well as control.

Massimiliano has developed AI and Internet of Things systems in civilian and military settings, led cybersecurity initiatives and contributed to international projects on edge computing and ethical AI. His technical path spans early operating systems such as DIGITAL UNIX and VAX environments to modern Linux platforms, from low-level optimisation to the development of complete applications. His languages range from Pascal and Fortran to Java and Python.

Above all, he considers his work a bridge between universities and industry, power and responsibility, code and meaning.

> "Writing code means giving shape to attention. Designing AI means choosing which questions we allow the machine to return to us as an echo."